


















Article

# Improving the Translation Environment for Professional Translators

Vincent Vandeghinste<sup>1,†</sup> , Tom Vanallemeersch<sup>1,‡</sup>, Liesbeth Augustinus<sup>1</sup> , Bram Bulté<sup>1</sup> , Frank Van Eynde<sup>1</sup> , Joris Pelemans<sup>1,§</sup>, Lyan Verwimp<sup>1</sup>, Patrick Wambacq<sup>1</sup> , Geert Heyman<sup>1,||</sup> , Marie-Francine Moens<sup>1</sup> , Iulianna van der Lek-Ciudin<sup>1</sup>, Frieda Steurs<sup>1,†</sup> , Ayla Rigouts Terryn<sup>2</sup> , Els Lefever<sup>2</sup> , Arda Tezcan<sup>2</sup>, Lieve Macken<sup>2</sup> , Véronique Hoste<sup>2</sup> , Joke Daems<sup>2</sup> , Joost Buyschaert<sup>2</sup> , Sven Coppers<sup>3</sup> , Jan Van den Bergh<sup>3</sup>  and Kris Luyten<sup>3</sup> 

<sup>1</sup> KU Leuven; firstname.lastname@kuleuven.be

<sup>2</sup> Ghent University; firstname.lastname@ugent.be

<sup>3</sup> Hasselt University; firstname.lastname@uhasselt.be

\* Correspondence: vincent@ccl.kuleuven.be; Tel.: +32-16-325089

† Current address: Instituut voor de Nederlandse Taal, Leiden

‡ Current address: CrossLang

§ Current address: Apple Inc.

|| Current address: Nokia Bell Labs

Version April 24, 2019 submitted to Preprints

**Abstract:** When using computer-aided translation systems in a typical, professional translation workflow, there are several stages at which there is room for improvement. The SCATE (*Smart Computer-Aided Translation Environment*) project investigated several of these aspects, both from a human-computer interaction point of view, as well as from a purely technological side. This paper describes the SCATE research with respect to improved fuzzy matching, parallel treebanks, the integration of translation memories with machine translation, quality estimation, terminology extraction from comparable texts, the use of speech recognition in the translation process, and human computer interaction and interface design for the professional translation environment. For each of these topics, we describe the experiments we performed and the conclusions drawn, providing an overview of the highlights of the entire SCATE project.

**Keywords:** computer-aided translation; machine translation; speech translation; translation memory-machine translation integration; user interface; domain-adaptation; human-computer interface

## 1. Introduction

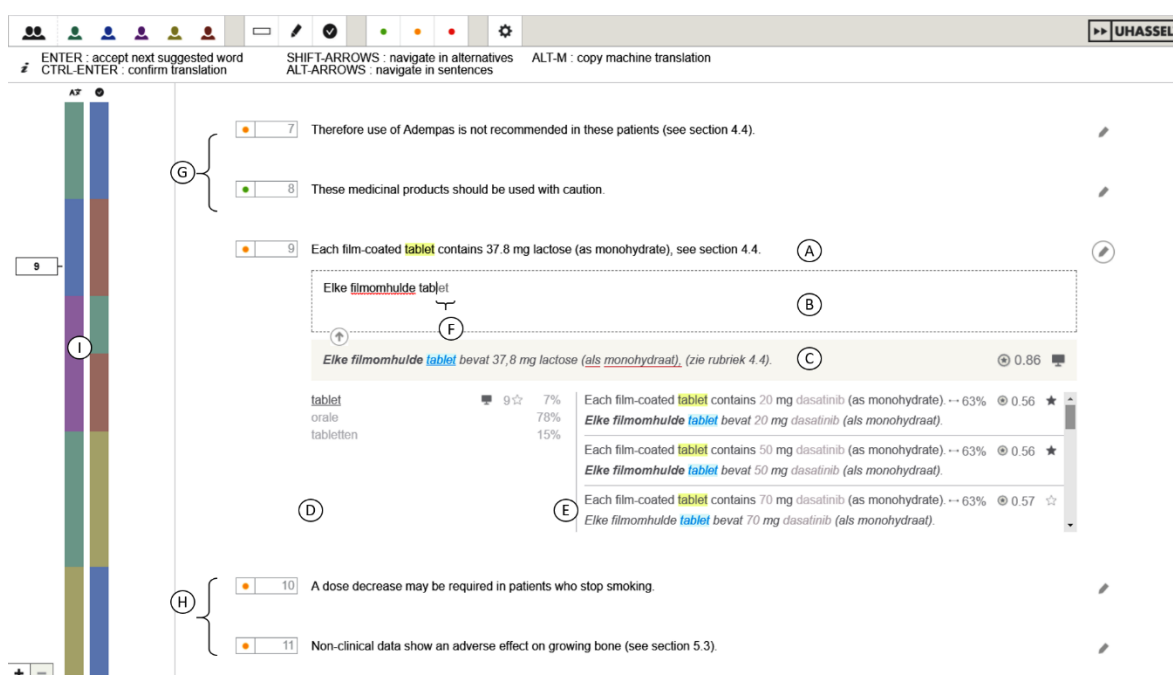
The SCATE project (*Smart Computer Aided Translation Environment*) was a four year research project that ran from March 2014 till February 2018, in which a consortium of three Flemish universities investigated several aspects and stages in the professional translation workflow, aiming at improvements in each of them. This paper describes the highlights of our research. Section 2 presents related work, whereas the remainder of the paper describes the work done within the SCATE project.

Figure 1 provides an overview of most parts of the project, through a prototype user interface (which is described in detail in section 7) of the *Smart Computer-Aided Translation Environment*.<sup>1</sup> A professional translates a sentence under translation (A) using a large text entry box centrally on the screen (B). The central placement provides space for context; (G) preceding and (H) subsequent sentences, overall translation progress (I) and configuration options in the top bar. Autocomplete (F) assists translators during their task. Suggestions come from multiple sources, all related to technologies

<sup>1</sup> A demo version of the prototype is available at <http://scate.edm.uhasselt.be/>.

26 developed or improved within the project. A translator can accept the default suggestion, choose an  
27 alternative term from the presented options (D) or start typing a different translation.

28 When starting the translation of a sentence, the default translation comes from hybrid machine  
29 translation. The complete translated sentence is presented immediately below the text box (C) and  
30 can be copied using a single shortcut. The hybrid machine translation builds on the research on both  
31 machine translation and fuzzy matching, discussed in section 3. As other results from fuzzy matching  
32 can help during translation, the top results are also presented to the translator in (E). At the right-hand  
33 side of both the hybrid machine translation (C) and fuzzy matches (E) quality estimations are presented.  
34 Research on quality estimation is described in section 4. The relevant terms of these fuzzy matches are  
35 also presented in the list of alternative (D), just as results from an automatically extracted term list, for  
36 which frequency information in the source is also presented. Results on the topic of term extraction are  
37 discussed in section 5. The integration of speech recognition in the translation process is described in  
38 section 6.



**Figure 1.** An overview of the SCATE interface. (A) The sentence to translate, (B) the editing field, (C) the hybrid MT that also includes pretranslations, (D) a list of translation alternatives coming from the term base, TM and MT, (E) fuzzy matches, (F) suggestion from autocomplete, (G) previous source sentences, (H) upcoming source sentences and (I) a progress bar.

## 39 2. Related Work

40 As translators rely on their computer-aided translation tools (CAT tools) to increase their  
41 productivity, end user satisfaction has become essential when developing new tools. Previous studies  
42 have shown that these aspects have been rather neglected in the past and the user interface design has  
43 been driven by the needs of the translation clients and not by the needs of the translator. [1,2]

44 Various surveys and field studies [3,4] investigating human-computer interaction, show that  
45 translators value improved translation memory (TM) - machine translation (MT) integration methods  
46 (e.g. copy/paste, drag-and-drop within editor). [5-7] show that reuse of sub-segments is possible  
47 through interactive translation prediction (ITP), a method in which users are presented, as they type,  
48 with translation suggestions from all available resources.

49 Suggestions are displayed either in a drop-down list or directly under the target segment.  
50 Translators seem to prefer ITP to classical post-editing because it minimises the number of keystrokes  
51 and thus increases productivity [8,9]. Commercial translation software developers have implemented

52 this technology in different ways and use different terminology to refer to it, such as *predictive typing*,  
53 *AutoSuggest*, *Autocomplete*, or *Autowrite*.

54 [10] shows that metadata can help translators make well-informed decisions. He concludes that  
55 metadata "helps translators adapt their translation strategies more easily according to the suggestion  
56 type". [4] indicates that translators like information about the provenance of the MT suggestions and  
57 estimation of their quality. In the context of post-editing, [11] argues that translators value on-the-fly  
58 highlighting of word alignment in order to keep the connection between source and target text. In  
59 other words, it appears useful to explicitly link parts of a source sentence with parts of the translation  
60 suggestion.

61 In SCATE we developed visual aids that explain the origin of the translation suggestions and  
62 their link with the source text.

### 63 3. Translation Technologies

64 Amongst the main translation technologies, besides a term-base (TB), that are accessible to most  
65 translators in their professional CAT environment are a TM system and an MT engine. Section 3.1  
66 describes how a TM system can improve the matching of existing translations with the segment to  
67 translate. Section 3.2 investigates integrating TM and MT technologies. Section 3.3 describes our efforts  
68 in the creation and accessibility of parallel treebanks (i.e. syntactically annotated parallel sentences)  
69 for syntax-based MT.

#### 70 3.1. Improved Fuzzy Matching

71 CAT tools have become indispensable in the environment of the modern translator. They help  
72 increase consistency, productivity and quality. One of the core components of a CAT tool is the TM  
73 system, which contains a database of already translated fragments, the TM. Given a sentence to be  
74 translated, the traditional TM system looks for source language sentences in a TM which are identical  
75 (*exact matches*) or highly similar (*fuzzy matches*), and, upon success, suggests the translation of the  
76 matching sentence to the translator.

77 Similarity calculation can be done in many ways. In current TM systems, fuzzy matching  
78 techniques mainly consider sentences as simple *sequences of words* and contain very limited linguistic  
79 knowledge, such as stop word lists. Few tools use more elaborate linguistic knowledge. We include  
80 *syntactic information* for detecting TM sentences which are not only similar when comparing words,  
81 but also when comparing the syntactic information associated with the sentences. Such information  
82 can consist of lemmas, part-of speech tags or syntax parse trees. We investigate whether such abstract,  
83 syntax-based matching is able to assess the usefulness of matches in a better way than methods purely  
84 based on sequences of words.

85 We designed a flexible and time-efficient framework which applies and combines different metrics  
86 in the source and target language. We measure the correlation of fuzzy matching metrics scores with  
87 the evaluation score of the suggested translation to find out how well the usefulness of a suggestion  
88 can be predicted, and we measure the difference in recall between fuzzy matching metrics by looking  
89 at the improvements in mean *Translation Edit Rate* (TER) [12] as the match score decreases.

90 Our comparison of the baseline matching metric, *Levenshtein distance* [13], with linguistically  
91 aware and unaware matching metrics, has shown that the use of linguistic knowledge in the matching  
92 process provides clear added value, especially when several metrics are combined into a new metric  
93 using a regression tree. The correlation of combined metrics with the evaluation score is much stronger  
94 than the correlation of the baseline. Moreover, there is significant improvement in mean evaluation  
95 score, and the difference in recall with the baseline increases as match scores decrease. Full details of  
96 this study can be found in [14].

97 The improved fuzzy matching system is implemented as a web service available through an  
98 application programming interface (API) and is used in the SCATE interface prototype, as shown in  
99 Figure 2. This prototype is discussed in more detail in Section 7.

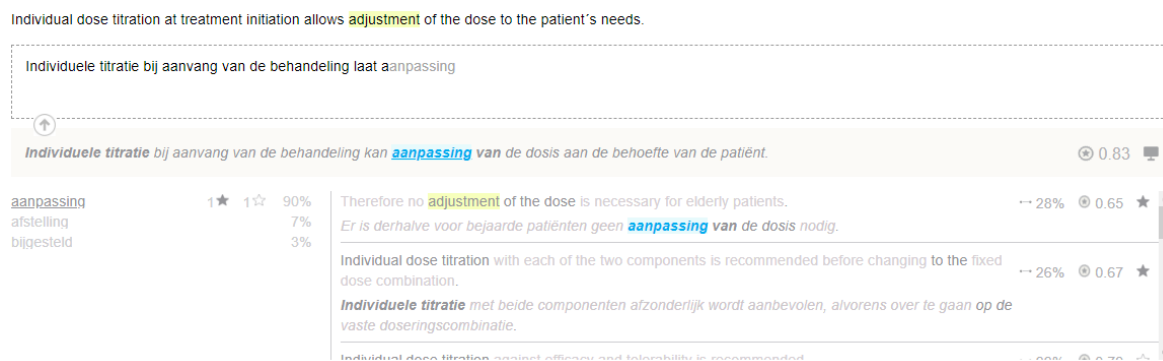


Figure 2. Fuzzy matches (bottom right) and integrated TM-MT suggestion (middle) in the prototype.

### 3.2. Integration of Translation Memory with Machine Translation

We test the integration of MT and TM, in order to increase the quality of and potentially the confidence in MT output. The TM-MT system consists of two main components: (1) fuzzy match repair, i.e. the automatic editing of close matches found in the TM, and (2) span pretranslation, in the context of which MT output is constrained by including certain consistently aligned subsegments coming from one or more TM matches. Both components use a TM with fuzzy matching techniques and a statistical MT (SMT) system with related alignment information. Different metrics are used for the retrieval and scoring of fuzzy matches, including the syntactic fuzzy matching metric described in section 3.1. We performed experiments on ten language pairs (English ↔ German, French, Hungarian, Dutch and Polish) which involve multiple language families, using the DGT dataset [15]. We applied phrase-based SMT without span pretranslation [16], pure TM and a recurrent neural network (RNN) encoder-decoder neural MT (NMT) system [17] as baselines, and evaluated the translations using several metrics. The tests show that this approach has potential. As shown in Figure 3, significantly higher BLEU scores [18] for nine of the ten language combinations were reported, and also METEOR [19] and TER [12] scores show comparable patterns. More details are available in [20]. The system is, as shown in Figure 2, also integrated in the SCATE prototype, which provides translators with informed MT output and which is described in section 7.

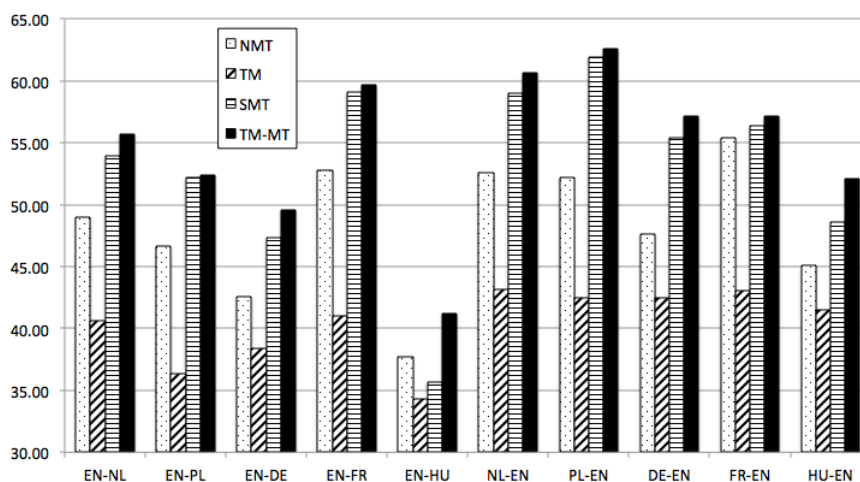


Figure 3. Overview BLEU scores TM-MT integration and baselines [20]

### 3.3. Parallel Treebanks

Parallel treebanks [21] are syntactically annotated versions of parallel corpora. While the latter are traditionally used in data-driven MT systems, such as *phrase-based SMT* or *NMT* [22], parallel

120 treebanks can be used to improve syntax-based statistical MT ([23], [24]) by taking advantage of  
121 linguistic information, allowing higher levels of abstraction than in phrase-based SMT.

122 Work on parallel treebanks also has potential to improve *tree*-based NMT, which is a very recent  
123 research topic. *Tree-to-string* approaches are, amongst others, described in [25], [26] and *string-to-tree*  
124 approaches in, amongst others [27] and [28]. While we are not aware of any *tree-to-tree* approaches in  
125 NMT (yet), we consider it only a matter of time before such approaches appear, as such techniques are  
126 already being used for e.g. computer program translation between different programming languages  
127 [29].

128 Below, we explain the concept of alignment (Section 3.3.1), leading to results like parallel treebanks,  
129 and the creation of MT rules from alignments (Section 3.3.2). We explain the SCATE work on enriching  
130 parallel treebanks with semantic information in order to bridge syntactic divergences and to facilitate  
131 MT rule creation (Section 3.3.3), and the work on allowing to search parallel treebanks (Section 3.3.4).

### 132 3.3.1. Sub-sentential Alignment

133 Alignment consists of linking segments of a source text with translation-equivalent segments of  
134 the target text, i.e. the translation of the source text. Starting at the *document level*, alignment is usually  
135 performed using an iterative refinement strategy. Alignment proceeds at the *sentence level*, and may  
136 continue at the *sub-sentential level* and the *word level*.

137 Sentence alignment is more or less considered a solved problem, at least for parallel documents.<sup>2</sup>  
138 Sub-sentential alignment consists of aligning elements below the sentence level, such as words, chunks  
139 or constituents at deeper levels of syntactic hierarchy. Word alignment deals with issues such as NULL  
140 links (untranslated words, or words added during translation), crossing links (changes of order of  
141 words during translation), and fuzzy links (e.g. translation of groups of words as a whole rather than  
142 as individual words). Word alignment in sentence pairs is typically produced using statistical tools  
143 such as GIZA++ [30], which also create a set of lexical probabilities based on the word alignments of a  
144 large set of sentence pairs. These probabilities indicate the likelihood a source word is translated by a  
145 target word or vice versa. The word alignment and lexical probabilities allow for the alignment of word  
146 groups, aligned groups being integrated into a so-called phrase table for SMT systems. Sub-sentential  
147 alignment may apply linguistic information by aligning chunks [31], which result from a superficial  
148 syntactic analysis of a sentence (detection of the boundaries of noun phrases and verb phrases), or by  
149 aligning nodes in parse trees, which provide a deep syntactic hierarchy of a sentence.

150 We focus on the alignment of nodes in syntactic parse trees (a.k.a. tree alignment), as this allows  
151 more flexible translation patterns for MT engines than mere word alignment. Several tree aligners  
152 exist ([32–34]) taking syntactic parse trees as input, using word alignments and lexical probabilities as  
153 input. Tree alignment leads to parallel treebanks. In other words, such treebanks [21] are syntactically  
154 annotated versions of parallel corpora.

### 155 3.3.2. Machine Translation rules

156 Based on alignment results, translation rules can be created. Data-driven MT systems such as  
157 phrase-based SMT [16] and NMT [17], at least in its standard form, use parallel corpora without  
158 annotations. Parallel treebanks, on the other hand, can be used to create syntax-based MT rules, and  
159 hence to develop syntax-based statistical MT systems ([23,24]). The linguistic information incorporated  
160 in their rules allows for higher levels of abstraction and more flexible patterns than the rules derived  
161 from non-annotated corpora. Figure 4 shows a sub-sententially aligned pair of parallel trees.

---

<sup>2</sup> <http://www.statmt.org/survey/Topic/SentenceAlignment> for an overview.

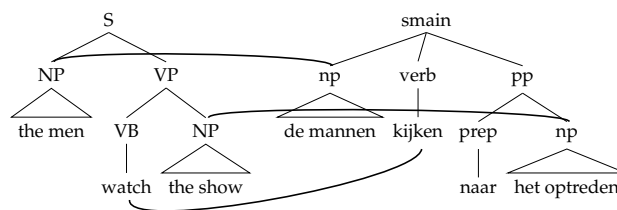


Figure 4. An example node-aligned parallel tree.<sup>3</sup>

162 The translation-equivalent sentences in parallel corpora may show syntactic divergences, i.e. use  
 163 different syntactic means to convey the same meaning as a result of linguistic necessities or translators'  
 164 choices. This makes alignment based on syntactic structure complex.

### 165 3.3.3. Semantic information

166 While the syntactic structure of sentences often changes during translation, semantic information  
 167 tends to remain constant. Therefore, we investigated whether aligning parse trees based on such  
 168 information facilitates alignment and leads to higher quality MT rules with respect to alignment purely  
 169 based on syntactic information. We focus on shallow semantics, in the form of predicates and roles.  
 170 We apply a five-step approach in order to obtain semantically motivated MT rules:

171 **Step 1:** Creation of a semantic role labeler. As tools for automatically assigning semantic predicates  
 172 and roles are scarce resources, we apply a crosslingual projection approach and train a semantic role  
 173 labeler from the projected information. We annotate syntactic parse trees in the resource-rich language  
 174 (English) with a semantic role labeler and project the predicate and role labels to the syntactic parse  
 175 trees in the target language (Dutch) through a non-linguistic tree aligner, LENG (Lexically Equivalent  
 176 Node Grouping) [35], which we developed in SCATE. Details of this aligner can be found below.

177 **Step 2:** From the projected labels, we train a semantic role labeler, requiring a minimum of manual  
 178 intervention. The labeler contains a model with mappings between syntax and semantics.

179 **Step 3:** We align parse trees via semantic labels, word alignment and lexical probabilities.

180 **Step 4:** We derive translation rules based on the aligned parse trees.

181 **Step 5:** We extend a phrase-based SMT system with the translation rules.

182 Evaluation results for step 3 and 5 indicate that enriching parse trees with semantic predicate  
 183 and role labels leads to more precise tree alignment results, and that combining a phrase table with  
 184 semantic translation rules helps in improving translation quality. While we performed tests on the  
 185 language pair English-to-Dutch, our approach is sufficiently generic for tests on other language pairs.  
 186 More details can be found in [35].

187 The LENG tree aligner, being non-linguistic, may also be applied in a broader context, beyond  
 188 semantically motivated MT. It combines the language pair and parser independence of [34] with the  
 189 higher performance of [33]. It looks for pairs of isomorphic source and target subtrees in which pairs  
 190 of nodes show a strong lexical equivalence. The tree alignment consists of linked subtree pairs that do  
 191 not overlap with each other. As opposed to [34], LENG does not only use lexical probabilities, but also  
 192 the word alignment of the sentence pair (similarly to [33]), imposes less well-formedness constraints,  
 193 and only links nodes to each other if there is strong evidence for doing so.

194 We compared LENG with [34] and [33] on the last 35 sentences in the 125-sentence Lingua-Align  
 195 gold standard, using the lexical probabilities and word alignment included with the gold standard.  
 196 Evaluation statistics are shown in Table 1. It shows that we clearly perform better than [34] on precision,  
 197 recall and F-score, and also outperform [33].

<sup>3</sup> Gloss of the Dutch sentence is "the men look at the show"

System	Precision	Recall	F-score
SubTree Aligner [34]	69.30	71.55	70.40
Lingua-Align [33]	79.29	88.78	83.77
LENG	<b>83.48</b>	<b>89.96</b>	<b>86.60</b>

**Table 1.** Subtree alignment accuracy on English-Dutch gold standard

### 198 3.3.4. Searching Parallel Treebanks

199 Parallel treebanks can not only be used for creating MT rules, but also as a resource for studying  
 200 translation phenomena. We built an updated version (with improved parses and improved alignment)  
 201 of the parallel Europarl treebank for Dutch and English [21]. This treebank is *tree aligned* (see also  
 202 section 3.3.1) and can be queried with Poly-GrETEL [36].

203 Poly-GrETEL is an online tool<sup>4</sup> which enables example-based syntactic querying in parallel  
 204 treebanks, and which is based on the monolingual GrETEL (Greedy Extraction of Trees for Empirical  
 205 Linguistics) environment [37]. The tool provides online access to the Europarl parallel treebank  
 206 for Dutch and English, allowing users to query the treebank using either an XPath expression or  
 207 an example sentence in order to look for similar constructions.<sup>5</sup> The treebank contains automatic  
 208 alignments between the nodes. By combining example-based query functionality with node alignments,  
 209 we limit the need for users to be familiar with the query language and the structure of the trees in the  
 210 source and target language, thus facilitating the use of parallel corpora for comparative linguistics  
 211 and translation studies. Poly-GrETEL will become part of the CLARIN<sup>6</sup> linguistic infrastructure for  
 212 researchers.

## 213 4. Quality Estimation of Computer-Aided Translation

214 Quality Estimation (QE) is defined as the task of providing a quality indicator for  
 215 machine-translated text without relying on reference translations. The aim of QE is to predict a  
 216 quality score at sentence and/or document level or more fine-grained error labels at word level that  
 217 indicate the need for post-editing. The general approach to QE consists of feature engineering, which  
 218 is the task of finding informative predictors (or features) of MT quality, and applying various Machine  
 219 Learning (ML) algorithms to build prediction models, which associate features with quality labels.

220 Today, despite their widespread adoption, ML models of QE remain mostly *black boxes*, where no  
 221 explanation for the predicted quality is provided [40–42]. In order to gain wide-spread acceptance,  
 222 besides building more accurate systems, one of the main challenges of QE can be considered to build  
 223 *white box* systems whose predictions can be justified. Based on the definition of the post-editing task,  
 224 one way of doing this would be to take a two-step approach, by detecting different types of MT errors  
 225 in the first step, which are then used in a second step to estimate a global score at sentence level. Such  
 226 systems would not only be beneficial for MT developers and end users to make a meaningful analysis  
 227 about the translation errors a certain MT system makes, but they can also yield higher productivity  
 228 gains in CAT workflows that utilise MT and can improve the acceptability of MT by post-editors, by  
 229 filtering out the sentences with the more challenging error types and by highlighting errors. In the  
 230 SCATE project, we use automatic error detection as a basis to two-step, informative quality estimation  
 231 systems for MT, which are able to justify the reasons for estimated quality.

232 In Section 4.1, we first describe a new taxonomy and annotated data set of MT errors. Section 4.2  
 233 describes our approach to building informative quality estimation systems.

<sup>4</sup> <http://gretel.ccl.kuleuven.be/poly-gretel/>

<sup>5</sup> Currently, this is limited to the years 2000 and 2001. After we speed up the process using [38] and [39], we expect to expand this to the entire Europarl corpus, version 7.

<sup>6</sup> Common Language Resources for Research Infrastructure, <http://www.clarin.eu>

#### 234 4.1. Taxonomy and Annotated Data Set of Machine Translation Errors

235 Despite the link between MT errors and post-editing effort, most QE systems predict overall  
 236 post-editing effort, without making a distinction between error types. Automatic error detection is  
 237 essential to build informative QE systems that are specialised in localizing different types of errors. To  
 238 this end, in Figure 5, we present the SCATE MT error taxonomy, a fine-grained, hierarchical taxonomy,  
 239 in which errors are classified according to the type of information that is needed to detect them. We  
 240 refer to any error that can be detected in the target text alone as a *fluency error*. Fluency errors are  
 241 concerned with the well-formedness of the target language, regardless of the content and meaning:  
 242 transfer from the source language. There are five main error subcategories under fluency errors:  
 243 *grammar, lexicon, orthography, multiple errors* and *other fluency errors*. *Accuracy errors*, on the other hand,  
 244 are concerned with the extent to which the source content and the meaning is represented in the target  
 245 text and can only be detected when both source and target sentences are analyzed together. Accuracy  
 246 errors are split into the following main subcategories: *addition, omission, untranslated, Do-Not-Translate*  
 247 *(DNT), mistranslation, mechanical, bilingual terminology, source errors* and *other accuracy errors*.

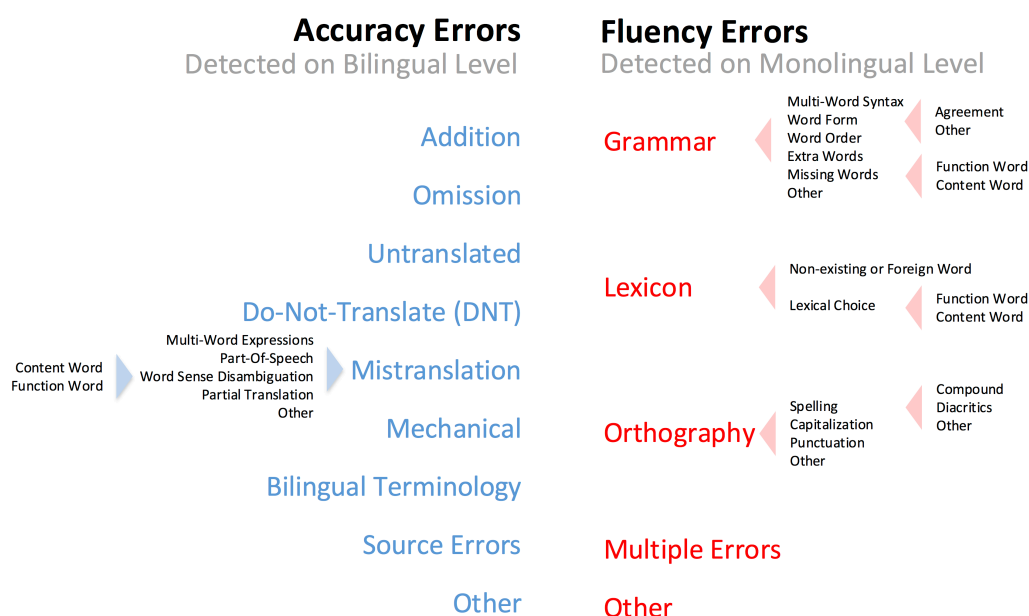


Figure 5. The SCATE MT error taxonomy

248 Certain similarities can be observed between some of the accuracy and fluency error categories  
 249 in the error taxonomy, such as *extra words* vs. *addition*, *missing words* vs. *omissions* or *orthography* -  
 250 *capitalisation* vs. *mechanical* - *capitalisation*. As the main distinction between accuracy and fluency errors  
 251 in the taxonomy is based on the type of information that is needed to be able to detect them, accuracy  
 252 errors do not necessarily imply fluency errors, or vice versa for that matter [43].

253 Using the SCATE MT error taxonomy, for the English-Dutch language pair, we built corpora of  
 254 MT errors consisting of output from three MT systems that are based on different MT paradigms:  
 255 SMT, Rule-Based MT (RBMT) and NMT. In these corpora of MT errors, we obtained error annotations  
 256 provided by multiple annotators, yielding high Inter-Annotator Agreement (IAA). We used Google  
 257 Translate (2014) as SMT system, Systran Enterprise Edition, version 7.5 as RBMT system and Google  
 258 Translate (2017) as NMT system to obtain MT output for all source sentences. The source sentences in  
 259 the corpus of SMT errors are extracted from the Dutch Parallel Corpus [44] and consist of an equal  
 260 number of sentences from three different text types: *external communication*, *non-fiction literature* and  
 261 *journalistic texts* (698 sentences in total). Furthermore, we extended the corpus of SMT errors (2,963  
 262 sentences in total) to analyze the relationship between MT error types and post-editing effort, and to



263 build automatic error detection systems, which are further explained in the next section. The details of  
264 the MT error taxonomy, the corpora of MT errors and the IAA analysis can be found in [43].

#### 265 4.2. Quality Estimation

266 We first discuss the predictive power of SMT errors in section 4.2.1, before discussing automatic  
267 error detection in section 4.2.2 and informative quality estimation in section 4.2.3.

##### 268 4.2.1. The predictive power of MT errors on temporal post-editing effort

269 From a post-editor's perspective, MT quality can be considered of the highest level when the MT  
270 system makes no serious translation errors, in other words when the effort required to post-edit is  
271 minimal. Despite the obvious relationship between the cognitive effort involved in post-editing and  
272 the translation errors made by the MT system, the impact and the predictive power of different types  
273 of MT errors on post-editing effort are yet to be fully understood.

274 With the hypothesis that the different error types an MT system makes can explain the cognitive  
275 effort involved in correcting them, we investigate whether ML techniques can be used to estimate  
276 Post-Editing Time (PET), an indirect measure of cognitive effort, by using gold-standard MT errors as  
277 features. We analyzed the SCATE corpus of SMT errors in combination with post-edits obtained for  
278 each MT output by two post-editors and the average PET calculated per sentence.

279 By using the gold-standard error annotations, we showed that PET can be estimated with high  
280 accuracy, provided that the types of errors in the MT output are known. We obtained these results by  
281 applying different ML techniques to the largest data set ever used in similar studies [45].

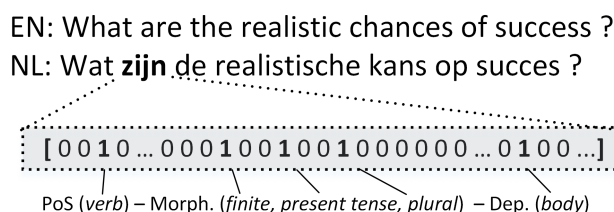
282 While these findings suggest that building two-step, informative quality estimation systems is  
283 possible in theory, accurate detection of all MT error types can be considered to be a challenging  
284 task, considering the different linguistic properties they represent. On the task of predicting PET, we  
285 applied various feature selection methods not only to seek a minimal subset of MT error types without  
286 reducing QE performance but also to reveal the predictive power of different error types on PET. Our  
287 results show that high QE performance can be achieved by using only eight error types (compared  
288 to all 33 error types) in the SCATE error taxonomy, corresponding to 31% of all gold-standard error  
289 annotations in the corpus. We observed the *Accuracy - Mistranslation* and *Fluency - Grammar* errors as  
290 two main error categories, whose sub-categories correspond to error types with high predictive power.  
291 Our findings suggest that we do not need to detect all error types to estimate PET successfully and  
292 error detection systems that focus only on error types with high predictive power on PET can lead to  
293 high quality sentence-level QE performances. For the details of our findings, we refer to [45].

##### 294 4.2.2. Automatic error detection

295 Considering the informativeness of the different types of MT errors on PET, we propose novel  
296 RNN architectures for word-level automatic error detection for *Fluency* and *Accuracy* errors as bases to  
297 building informative QE systems for predicting PET on sentence level.

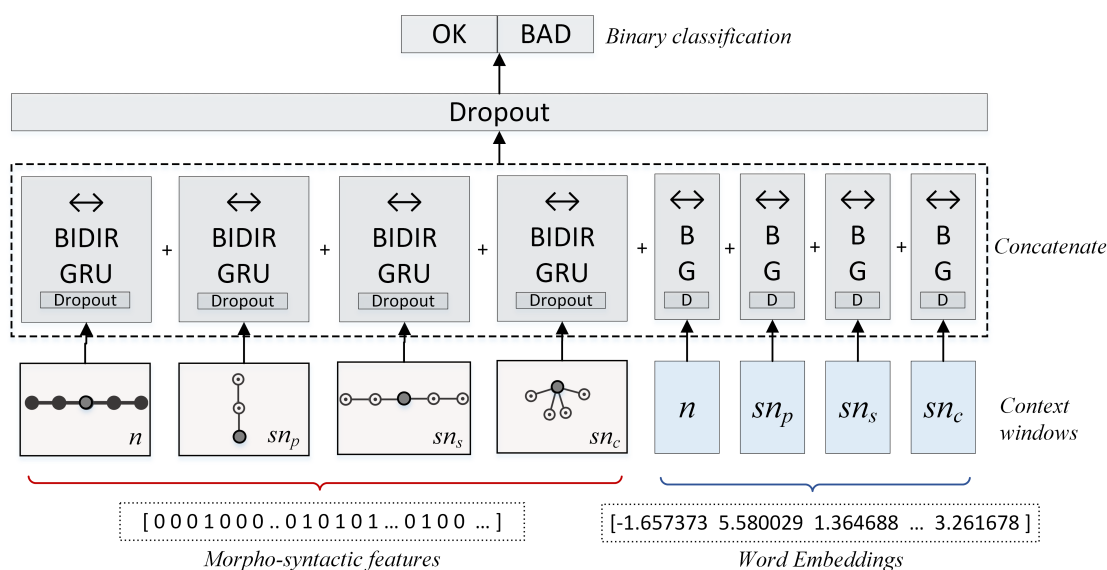
298 In order to train Neural Networks (NNs) on the task of detecting fluency errors, which are  
299 concerned with the well-formedness of the target text alone, we propose a new word representation  
300 method, in which we transform each word in a given MT output into a feature vector using multi-hot  
301 encoding, which consists of three types of information: Part-of-Speech (PoS), morphology and  
302 dependency relation, which we extract by using the Alpino dependency parser for the Dutch language  
303 [46]. In each word vector, all elements are assigned the value of 0, except the elements representing the  
304 linguistic features of each word, which are assigned 1. Unlike word embeddings, the morpho-syntactic  
305 representation strips out semantic features from words. One difficulty of using dependency parsing  
306 on MT output is that the generated parse trees can be unreliable when the MT output itself contains  
307 errors. On the other hand, multiple studies demonstrated that parse trees obtained on MT output  
308 nevertheless provide useful information in terms of MT quality [47,48].

309 Figure 6 shows an example source sentence (EN), its machine-translated version (NL) and the  
 310 morpho-syntactic representation for the word *zijn* (*are*). The MT output in this figure contains a *Fluency*  
 311 - *Grammar* error in the form of subject-verb agreement in number between the words *zijn* (*are*) (plural)  
 312 and *kans* (*chance*) (singular).



**Figure 6.** Binary vector for *zijn* (*are*) consisting of 1s for its PoS, morphology and dependency features and 0s for the remaining items in the vocabulary.

313 Besides surface context windows ( $n$ -grams), we utilised syntactic context windows for each given  
 314 target word, which we extracted from the dependency parse tree for each given MT output. Syntactic  
 315  $n$ -grams enable us to capture long-distance dependencies in MT output, which can be considered  
 316 as an important piece of information especially for detecting *Fluency - Grammar* errors. Combining  
 317 morpho-syntactic features with surface and syntactic  $n$ -grams, we propose an RNN architecture, which  
 318 is illustrated in Figure 7.



**Figure 7.** The proposed neural network architecture for detecting fluency errors. While  $n$  represents a surface  $n$ -gram,  $sn_p$ ,  $sn_s$  and  $sn_c$  represent syntactic  $n$ -grams obtained around the target word by considering its parents, siblings and children as context in a given dependency tree.

319 In the proposed RNN architecture, we provide morpho-syntactic feature vectors and  
 320 word-embedding vectors of a target word in the form of surface and syntactic  $n$ -grams into eight  
 321 parallel Gated Recurrent Unit (GRU) layers, whose output is concatenated before they are connected  
 322 to the output layer. This network, as a result, predicts if a given word contributes to a grammatical  
 323 error or not as a binary classification task.

324 Our findings showed that the combination of the two types of information achieved better QE  
 325 performance on the task of detecting all fluency errors, than using either type of information as  
 326 input in isolation. Moreover, on the task of detecting *Fluency - Grammar* errors in SMT output, we  
 327 achieved a marked improvement in performance by using accurate morpho-syntactic features over  
 328 word-embeddings.

329 To detect accuracy errors, we modify the proposed RNN architecture and instead of using  
 330 morpho-syntactic features of the target text, we use word-embedding information obtained on the  
 331 source and target texts as input. Our approach additionally incorporates automatic word alignment  
 332 techniques to extract relevant information from the source text. We show that the proposed method  
 333 achieves the best results compared to other NN configurations that utilise morpho-syntactic features  
 334 as additional input. For the details of our experiments on automatic error detection, we refer to [49].

### 335 4.2.3. Informative quality estimation

336 Automatic error detection of fine-grained error categories remains a highly challenging task.  
 337 However, the predictions obtained from the error detection systems on more coarse-grained error  
 338 categories, such as dedicated systems for all accuracy and all fluency errors perform relatively well and  
 339 serve as valuable features for building informative QE systems to predict PET. Furthermore, additional  
 340 experiments show that the predictive power of such informative sentence level QE systems could  
 341 be maximised with additional sentence-level features obtained on a given source/MT output pair,  
 342 yielding 96% of the Pearson's correlation score of the upper boundary we observed on this task by  
 343 using gold-standard error annotations as features [49].



Figure 8. Quality estimation output in the SCATE user interface.

344 One of the aims for building informative QE systems is to inform the users about the reasons  
 345 for the estimated quality. Figure 8 shows how informative QE is presented to the user on the  
 346 SCATE platform. Words that are underlined in red are the words that correspond to fluency errors,  
 347 which are detected automatically. The score to the right of the MT output (0.56) corresponds to the  
 348 predicted sentence-level quality, which is based on the output of word-level error detection systems  
 349 and additional sentence-level features, calculated as  $1 - TER$ .<sup>7</sup> As illustrated in this figure, the SCATE  
 350 platform not only highlights the type and location of errors in a given MT output but also uses this  
 351 information to predict its sentence-level quality.

352 Even though predicting the exact location of MT errors remains a challenging task, we observe  
 353 that the proposed systems approximate the location of errors with greater success. Moreover, despite  
 354 the given challenges, our findings confirm that using automatic error detection systems as a basis for  
 355 sentence-level QE is a promising approach to build informative QE systems. We demonstrate that the  
 356 proposed methods deliver QE systems that perform well on estimating temporal post-editing effort,  
 357 while providing meaningful predictions about the type and location of the translations made by a  
 358 given MT system. For further details, see [49].

## 359 5. Terminology Extraction

360 We first describe our observations of translator's methods for acquiring terminology (Section 5.1),  
 361 before we describe our approach to automatic term extraction from comparable text (Section 5.2).

<sup>7</sup> In this example, sentence-level quality is measured in terms of technical post-editing effort as we did not have PET information on this data set.

### 362 5.1. Studying translator's methods of acquiring domain-specific terminology

363 To identify translators' terminology strategies of acquiring new domain knowledge, we launched  
364 an online questionnaire and visited language professionals at their workplaces. The questionnaire  
365 contained a total of 46 questions out of which 13 concerned demographics and professional experience,  
366 9 concerned the translation work environment, and 9 concerned terminology activities. The  
367 questionnaire was answered by 187 language professionals worldwide, out of which more than 70%  
368 were freelance translators and the rest were in-house translators/revisers, terminologists, interpreters,  
369 post-editors, and project managers. The questionnaire was online between December 2014 and  
370 February 2015.

371 In the field, we observed 13 translators and 3 terminologists in their authentic professional work  
372 environment (freelance, commercial and institutional settings) by applying the Contextual Inquiry [50],  
373 and Think Aloud Protocol (TAP) [51] research methods. The workplace visits took place in Belgium,  
374 the Netherlands and Luxembourg and were spread over a period of 6 months between November  
375 2014 and June 2015. For more details we refer to [3].

376 The study reveals information about translators / terminologists' terminology acquisition and  
377 management practices, web search behaviour and usage of online linguistic resources to solve  
378 terminological problems. Out of 187 survey respondents, about 139 indicated performing terminology  
379 activities. About 88% collected terms manually, while 22% used semi-automatic term extraction  
380 programs. More than half (about 52%) stored their terms in their CAT termbase, while 43% in  
381 a spreadsheet. The rest preferred a text processor (27%) and standalone translation management  
382 systems (15%). More than half stored only the language equivalents in their termbases. As for term  
383 research activities, the online resources were most exploited, followed by personal resources and  
384 client's resources. Finally, the survey helped us identify needs and shortcomings of the terminology  
385 management component integrated in CAT tools, related to the integration with online databases and  
386 exchange of terminological data. For more information see [52].

387 During the contextual inquiries at translators' workplaces we noticed the following types of  
388 terminology problems that occurred during translation:

389 (1) Related to specialised terminology: the translator does not know the meaning of the source  
390 term; the translator does understand the source term but does not know how to translate it in the  
391 target language; the translator does not know which target language equivalent to select from several  
392 translation alternatives coming from a large database.

393 (2) Related to general language.

394 (3) Related to the translation of named entities, acronyms, ambiguity, low quality of the source  
395 text, and punctuation.

396 To find a solution, translators used various tools, search and retrieval strategies both from local  
397 and online resources. We summarise the main findings below:

398 Both the survey and the field observations revealed that translators rely more on their TMs than  
399 on termbases to retrieve translation solutions. When no matches are found, the translator can perform a  
400 bilingual concordance search, in which the source term is highlighted and a target sentence is shown as  
401 such, with no highlight of the translation equivalents. The translators has to copy/paste the preferred  
402 translation from the concordance result window into the target sentence. We saw that the concordance  
403 feature was the second preferred CAT tool feature, after the TM match retrieval functionality. The  
404 over-reliance on TMs is signalled and discussed in early studies as well, e.g. [53,54]. While parallel  
405 corpora can be very useful to analyse translation equivalents in their context, [55] warns that they can  
406 have a major drawback in the fact that "they require the existence of a translation history" and they are  
407 not "faithful to linguistic uses in the target language." She further emphasises that comparable corpora  
408 (collections of original texts in two or more languages assembled on the basis of similarity) can also be  
409 a good alternative to acquire specialised knowledge and terminology for under-resourced languages  
410 and emerging fields. Despite its proven usefulness [56] the SCATE survey shows that comparable

411 corpora are hardly exploited for terminology and knowledge acquisition, the only resource mentioned  
412 being Wikipedia. SketchEngine that contains the TenTen Corpus Family<sup>8</sup> was mentioned only by one  
413 participant out of 139 who indicated performing terminology activities.

414 Besides the concordance feature, the translator can also use the term extraction feature  
415 incorporated in their CAT tool to quickly retrieve term candidates from their TMs and reference  
416 corpora, validate the term and add them to their termbase for future use. Most tools incorporate a  
417 monolingual term extraction component, whereas our research shows that there is also a need for  
418 bilingual and multilingual automatic term extractors. In addition, the survey showed that only 19  
419 of a total of 187 used the term extraction feature in their CAT tool. Some reasons for the low usage,  
420 revealed during the observations: the users did not know how to configure the extraction parameters  
421 and the validation of the term candidates was time-consuming due to the amount of noise.

422 Besides TM, the institutional translators also had access to a custom MT system that they could  
423 used to retrieve possible translation suggestions for terms, phrases or entire segments when there were  
424 no matches coming from the TMs. None of the commercial translators we observed used MT via the  
425 plugins integrated in their CAT tools.

426 Another method to search for terminological information or translation equivalents is to look  
427 up terms and phrases in external databases directly from the CAT tool's translation editing interface.  
428 Although most translation environments offer look-up functionality in external terminology databases  
429 (e.g. IATE, UnTerm, EuroTerm) and parallel corpora (e.g. MyMemory), our research shows that the  
430 integration with CAT tools is not optimal. Both commercial and institutional translators indicated  
431 that more advanced filtering techniques are required in order to query the IATE database directly  
432 from the CAT tool's interface. In addition, online databases are not always up to date, may contain  
433 outdated references, or may reflect the terminology used by a specific organisation. Nevertheless,  
434 things have changed since the study finished. The IATE team has launched a new version of IATE that  
435 is user-friendlier. Recently, in a JIAMCATT local meeting,<sup>9</sup> it was announced that SDL was developing  
436 a plugin for IATE to allow translators search the database directly from the interface of SDL Trados  
437 Studio.

438 When the local resources did not return any useful results for terminology and translation  
439 problems, the translators switched to the Web to look for a solution by consulting various websites,  
440 online dictionaries and platforms. Similarly to the results of the TTC survey [57], both our survey and  
441 field research revealed that online resources are the most popular linguistic resource for researching  
442 terminology. Figure 9 shows the most used resources from each category.

---

<sup>8</sup> <https://www.sketchengine.eu/documentation/tenten-corpora/>

<sup>9</sup> JIAMCATT is the International Annual Meeting on Computer-Assisted Translation and Terminology. JIAMCATT membership includes most international organizations, as well as various national institutions and academic bodies, active in the field of terminology and translation.

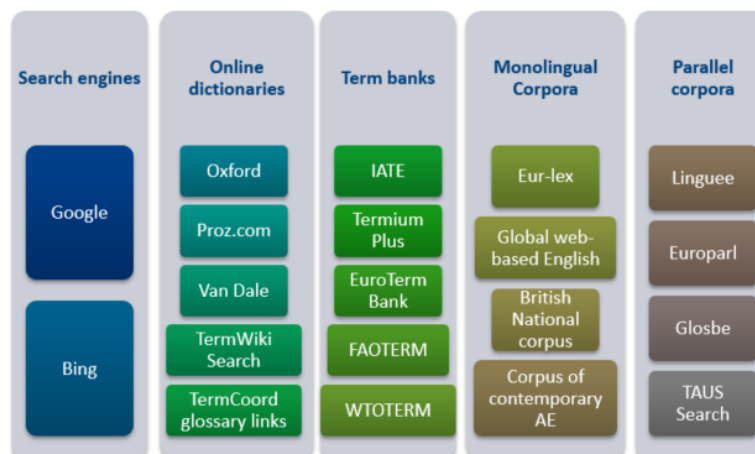


Figure 9. Most used online terminological resources

443 During the observations, we noticed lots of back and forth switching between several types of  
 444 online resources before taking a final decision. The web research path was often decided by the number  
 445 of hits Google gave with the web searches resulting in desktop clutter as the user did not know how  
 446 to manage the search results. For this purpose, one of our subjects developed a strategy not to keep  
 447 more than three tabs open on his desktop. He also used the Ditto clipboard manager to record his  
 448 searches, which saved time. Though the Google search engine was often used, out of the 16 translators  
 449 we observed only 2 used some advanced search operators in Google. When a translation solution was  
 450 found, it was copied/pasted in the translation grid and confirmed in the Translation Memory. Useful  
 451 websites were added to the Favourites toolbar. At the European Parliament, for example, the web  
 452 links were usually centralised and shared via the internal portals of the terminology and translation  
 453 units. Out of 16 observations, we noticed only one instance when the translator actually stored the  
 454 information about the researched term in their term base. These findings correlate with the results of  
 455 the survey that revealed the reasons why translators do not perform proper terminology management:  
 456 lack of knowledge about terminology management, someone else's responsibility, no added-value,  
 457 time consuming, termbases are complex.

458 Another method of acquiring domain-specific terminology is manual compilation of small  
 459 thematic corpora with materials collected from the Web, which can be followed by manual term  
 460 extraction of a list of term candidates, validation, and import of the final terms into the terminology  
 461 database. The source term entries are then researched and completed with target-language equivalents.  
 462 This practice was observed during the observations of the 3 institutional staff terminologists. While the  
 463 manual collection of the corpora and extraction of terms are reliable methods of harvesting terminology,  
 464 the participants indicated that it was time-consuming. Ideally, the users should be able to collect  
 465 corpora automatically and query directly from their translation environment tool. Although there are  
 466 standalone corpus compilation and query tools, such as Sketch Engine, BootCat, AntCont, the SCATE  
 467 survey shows that they are hardly known and used by translators. This might be due to the fact that  
 468 such tools are not supported in the CAT tool. In 2016, Sketch Engine developed a plugin for SDL  
 469 Trados Studio to enable translators and terminologists to perform searches in their large collections of  
 470 corpora (e.g. EurLex) directly from the Translation Editing interface. The pilot showed that the plugin  
 471 was hardly used by translators and, therefore, further development stopped.

472 Overall, the field research confirms the finding of previous studies that terminology management  
 473 is mainly done on an *ad hoc* basis due to time pressure, lack of resources, limited knowledge of how to  
 474 manage terminology properly, and lack of immediate financial compensation. A systematic approach  
 475 was observed only at the European Institutions and large commercial organizations which had a  
 476 dedicated terminologist in each translation unit. Translators seem to rely heavily on their specialised  
 477 TMs rather than on termbases and/or comparable corpora. Semi-automatic term extraction, though  
 478 an integrated component in the commercial CAT tools, has not yet become a standard practice in

479 the preparation stage of a translation project. The Web represents a rich resource for knowledge and  
480 terminology acquisition but very few adopted the automatic tools for corpora compilation and query.  
481 Finally, more efficient web search strategies are needed in order to avoid desktop clutter and save  
482 and store the relevant information in an efficient way. The findings have implications for translators  
483 educators and software developers alike.

484 One way of optimizing the exploitation of external linguistic resources for the purpose of  
485 terminology acquisition is a seamless integration of more sophisticated terminology extraction methods  
486 from comparable corpora.

## 487 5.2. Terminology extraction from comparable text

488 We experimented with three types of comparable corpora. The first type are corpora compiled  
489 from Wikipedia articles, which are a valuable resource for compiling comparable corpora. Wikipedia  
490 articles have the benefit that they are annotated with the categories they belong to as well as with  
491 *interwiki* links, which link an article to its counterparts in other languages. Both types of annotations  
492 allow easy compilation of a comparable corpus that is both domain-specific (using the category  
493 labels) and strongly comparable across languages (using the interwiki links). For our experiments,  
494 we constructed an English-Dutch comparable corpus in the medical domain, containing about 1000  
495 document pairs. Datasets with aligned Wikipedia articles can be found online for many language pairs  
496 on the website of linguatools.<sup>10</sup>

497 The second type are corpora compiled from Reuters news articles. News articles are another  
498 resource to create comparable corpora. We experimented with the Reuters news dataset,<sup>11</sup> a  
499 multilingual collection of news articles published within the same time span. From this collection,  
500 we created a weakly-comparable corpus by comparing the topic labels (e.g., *global*, *economy*, etc.) that  
501 are annotated on the Reuters documents, for example: when an English document and a Spanish  
502 document are both annotated with the same global label they are considered to have comparable  
503 content and are added as a document pair to the comparable corpus. We analysed the resulting dataset  
504 with multilingual probabilistic topic models: *Bilingual Latent Dirichlet Allocation (BiLDA)* [58] and  
505 *Comparable Bilingual Latent Dirichlet Allocation (C-BiLDA)* [59]. We found that, although the C-BiLDA  
506 model could uncover some interesting cross-lingual topics (clusters of related words), the dataset was  
507 not well-suited for inducing translations as the domain was too broad and the comparability across  
508 languages too low. We therefore conclude that to construct comparable corpora from news articles  
509 merely relying on high-level topic labels is insufficient. Other clues like named entities (persons,  
510 locations) and publication dates should be taken into account.

511 The third type are existing comparable corpora. Several automatically crawled and cleaned  
512 comparable corpora have been made freely available online in the context of the TTC project.<sup>12</sup> These  
513 are all specialised corpora in specific domains, such as wind energy and mobile technology. They are  
514 available in different formats and in seven languages: English, French, German, Spanish, Russian,  
515 Latvian and Chinese. These characteristics make the corpora especially suited for experiments with  
516 automatic term extraction from comparable corpora. An additional advantage is that there are also  
517 (very) limited, manually validated reference term lists available for the evaluation of monolingual  
518 automatic term extraction. A final advantage is that the corpora have been used in previous  
519 experiments with automatic term extraction from comparable corpora, so any new results can easily  
520 be benchmarked against the state of the art.

521 We split cross-lingual terminology extraction into two subproblems: (1) term extraction, the  
522 identification of which words and phrases are (in-domain) terms ; and (2) term linking, where the

---

10 <https://linguatools.org/tools/corpora/wikipedia-comparable-corpora/>

11 <http://trec.nist.gov/data/reuters/reuters.html>

12 <http://www.lina.univ-nantes.fr/?Linguistic-Resources-from-the.html>

523 aim is to link terms to their correct translation. We focus mainly on term linking. We investigate  
524 word-level methods for bilingual lexicon induction (BLI), the task of finding translations for words  
525 and phrases from non-parallel texts; we propose a novel BLI model that integrates character-level and  
526 word-level representations; and we implement a hybrid compound splitter for Dutch that combines  
527 corpus frequency information with linguistic knowledge.

### 528 5.2.1. Comparison of weakly-supervised word-level BLI models

529 During the course of the project, we saw the rise of word embeddings in natural language  
530 processing. These vector representations have shown to encode useful syntactic and semantic  
531 properties of words and have also been used to build cross-lingual spaces where translations are  
532 mapped to similar representations. Most techniques that build such cross-lingual representations  
533 require parallel corpora or bilingual dictionaries, however. We study approaches that can learn  
534 cross-lingual representations without the need for an initial seed dictionary.

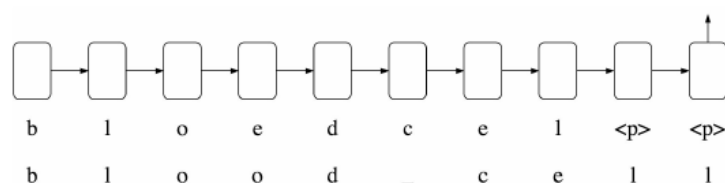
535 In particular, we compare two bilingual topic models, BiLDA and C-BiLDA, with a bilingual  
536 extension of the continuous skip-gram model called *Bilingual Word Embedding Skip Gram (BWESG)* [60].  
537 All three models learn bilingual word representations from subject-aligned document pairs only.  
538 Multilingual topic modeling has shown to be a robust framework for learning bilingual representations  
539 from such non-parallel data: BiLDA has been successfully applied to BLI [61] and C-BiLDA is a more  
540 recent extension to BiLDA that learns higher-quality representations when the aligned document pairs  
541 exhibit a lower degree of parallelism [59]. BWESG is a simple but effective extension to continuous  
542 skip-gram. It merges each aligned-document pair in a single bilingual document and then runs  
543 monolingual skip-gram with negative sampling [62] on the resulting document collection. To evaluate  
544 the models, we use a corpus of subject-aligned Wikipedia documents (English-Dutch) in the medical  
545 domain. From the English side of the corpus we selected 500 words, which were translated into Dutch  
546 to form the ground truth. We found that BWESG yields the best performance which indicates that also  
547 in a weakly-supervised settings, without parallel data, word embeddings are important BLI features.

### 548 5.2.2. Combining word-level and character-level representations

549 From our word-level experiments, we observe that for our dataset (consisting of Wikipedia articles  
550 in the medical domain) morphology is an important clue for identifying translations. Most recent  
551 work in BLI focuses solely on word-level features, however. For this reason, we design a model that  
552 seamlessly integrates word-level features (e.g., continuous skip-gram embeddings) and character-level  
553 features. Most related work in bilingual lexicon induction manually defines a cross-lingual similarity  
554 metric between word feature vectors. For instance, many methods use cosine distance to measure the  
555 similarity between embeddings. It is not trivial to define a similarity metric that incorporates both  
556 word- and character-level information, however. Therefore, we frame bilingual lexicon induction as a  
557 classification problem. We train a binary classifier that predicts whether two given words are each  
558 other's translation. The classifier's parameters are learned from a seed lexicon of known translations.

559 We identify two key advantages of a classification framework for BLI. Firstly, it does not rely on  
560 an *ad hoc* combination of features, but learns the patterns over different features from the bilingual seed  
561 lexicon. Secondly, the classification framework enables learning useful character-level features from  
562 the seed lexicon. This in contrast to using handcrafted features like normalised edit distance. In our  
563 model, we obtain a character-level representation by feeding the concatenation of source and target  
564 characters to an LSTM network (see Figure 10). As word-level representations, we used continuous  
565 skip-gram word embeddings. The concatenation of word and character features serves as the input to  
566 a feed-forward neural network that outputs a score between 0 and 1. The higher the score, the more  
567 confident the model is that the two given words are each other's translations.





**Figure 10.** Character-level representation in an LSTM framework

568 Our experiments show that the LSTM representation outperforms handcrafted morphology  
 569 features like normalised edit distance. Furthermore, the model that combines character-level  
 570 information and word-level information outperforms other baselines (including BWESG, the strongest  
 571 word-level model) by a margin. For more details, see [63].

572 In follow-up work [64], we verify that we can extend the BLI system, which could only find  
 573 translations for single words, to deal with phrases. Specifically, we find that, after extracting phrases  
 574 using a simple data-driven heuristic, we can treat phrases as if they were a single word: To learn  
 575 character-level representations, we treat whitespace as any other character, and to learn word-level  
 576 representations, phrases are tokenised as a single token.

### 577 5.2.3. Datasets and Gold Standards for Future Research

578 Finding comparable corpora for bilingual term extraction is not easy. Wikipedia is a useful  
 579 resource, but for very specialised subjects or smaller languages, coverage is not always optimal.  
 580 Moreover, while the strong comparability per document is useful, it is rare in other resources.  
 581 Compiling comparable corpora *ad hoc*, such as the one from Reuters new articles, is convenient,  
 582 but still requires identification of the terminology. Finally, a few comparable corpora are available with  
 583 manual term annotations, such as the one used from the TTC project. However, these are very rare  
 584 and often contain only monolingual annotations or a very limited list of cross-lingual links. This lack  
 585 of good resources means that evaluation can be challenging, and it is an important obstacle for the  
 586 development of supervised ML approaches for both monolingual and multilingual term extraction  
 587 from comparable corpora.

588 To address this, we started building a dataset for term extraction, which can be used both as  
 589 a gold standard and as training data for a supervised ML approach. To ensure re-usability of the  
 590 data, we collect corpora in three different languages (English, French, and Dutch) and four domains  
 591 (corruption, dressage, heart failure, and wind energy). These corpora are partly based on previous  
 592 research (e.g. the wind energy corpus uses the French and English parts of the TTC corpus). In each  
 593 corpus, around 50,000 tokens are manually annotated, using an annotation scheme with three different  
 594 term labels and elaborate guidelines. The guidelines, including information about the term labels, are  
 595 freely available online.<sup>13</sup> This results in a total of over 100,000 manual annotations in all corpora. We  
 596 are currently experimenting with an ML approach to term extraction based on these data.

597 While this is already a useful resource, as explained in the previous sections, multilingual term  
 598 extraction from comparable corpora involves two tasks: identifying terms and linking equivalent  
 599 terms across languages. Since the described data only addresses the former, more annotation work  
 600 was required to provide data for the latter. Therefore, the trilingual corpus about heart failure was  
 601 selected and both inter- and intralingual links between terms were manually identified: equivalents  
 602 across languages, synonyms, abbreviations, alternative spellings, hypernyms, hyponyms etc. In total,  
 603 7,385 unique terms and named entities in three languages were annotated this way. This dataset  
 604 is particularly suited as a gold standard for multilingual term extraction from comparable corpora

<sup>13</sup> <http://hdl.handle.net/1854/LU-8503113>

605 for two reasons. First, the inclusion of information about related terms means that a more nuanced  
606 evaluation can be made in cases when the automatically suggested target language term is not exactly  
607 an equivalent of the source term, but is still strongly related. Second, the fact that all terms have  
608 been annotated in this corpus means that the origin of wrongly suggested equivalents can be traced:  
609 either the system could not find the equivalent, or it was not present in the corpus. After all, since  
610 comparable corpora are not aligned, it is not unusual for a term to exist in one language of the corpora  
611 and not the other.

612 While these datasets could not yet be used to evaluate the systems presented in the previous  
613 section, they have already proven to be valuable sources of information about both terminology and  
614 comparable corpora [65]. For instance, the lack of restriction about length or part-of-speech of the  
615 terms revealed that, as expected, nouns and noun phrases are most common, but that, somewhat  
616 surprisingly, other part-of-speech patterns were often identified as well, e.g. adjectives and even  
617 verbs. Single-word and two-word terms appeared most often, but longer terms, up to around five  
618 tokens, were no exceptions. Ongoing research will have to confirm the further use of the data for the  
619 development of new tools. The dataset will be made available through a shared task on supervised  
620 machine learning approaches for automatic term extraction in 2020.

## 621 6. Speech Recognition

622 In the context of post-editing, using speech instead of typing can speed up the work of the  
623 translator. The accuracy of automatic speech recognition (ASR) can be improved by making use of the  
624 extra information present in the translation model (Section 6.1) and by adapting the language model  
625 to the current domain or topic (Section 6.2). Additionally, we explore the challenging task of speech  
626 translation in Section 6.3.

### 627 6.1. Adaptation of the Speech Recognition Language Model by Machine Translation

628 The aim of this research is to employ improved language models (LMs) and achieve higher  
629 recognition accuracy for spoken translations. We investigate two ways of improving the LMs: (1) using  
630 word translations to cluster similar words, which improves the reliability of word frequency statistics;  
631 (2) using the source language text and MT probabilities to steer the recognition in the right direction.

632 The first approach assumes that two words are similar, both semantically and syntactically, if they  
633 share the same translation in multiple languages. Similar words can then be clustered, which enables  
634 context sharing within each cluster and hence more reliable statistics for  $n$ -gram LMs containing these  
635 words. By filtering out translation errors based on part-of-speech, context and morphology, we are  
636 able to derive meaningful synonym clusters, but this does not result in improved recognition, mostly  
637 due to context insensitivity, i.e. words may be synonymous in certain contexts, but not in others.

638 The second approach investigates how to improve speech recognition, based on the source  
639 language text and MT probabilities. Research in the past largely focused on rescoring either ASR  $n$ -best  
640 lists or word lattices, using the MT probabilities of the source language text. This has the disadvantage  
641 that it requires two steps, which slows down recognition and requires intermediate storage. Moreover,  
642 such multi-pass approaches are often inferior to integrated approaches because information that is lost  
643 during the first step can never be recovered in the second step. Therefore we focus on integrating the  
644 source language text and MT probabilities into the LM directly. By weighing the  $n$ -gram probabilities  
645 with the translation probabilities of the source language text, a new LM can be created for each  
646 sentence/paragraph which can directly be used by an ASR decoder. This implementation allows to  
647 reduce recognition errors by ca. 5% absolute and 20% relative on spoken Dutch translations from  
648 English, while having little to no negative effect on recognition time. Moreover, compared to an  
649 existing model [66], our model takes up only 2.8% of disk space compared to a normalized model and  
650 dramatically reduces the execution time. More information can be found in [67].

651 Although the above implementation drastically improves the efficiency of MT-based LM  
652 adaptation, it assumes that translation consists solely of one-to-one alignments i.e. each word in

653 the source language text can only correspond to one word in the target language text. This is a strong  
654 assumption that does not hold in reality: every language has its own way of verbalizing concepts with  
655 some using a single word and others using multiple words for the same concept. In MT this issue is  
656 addressed by phrase-based translation models.

657 We integrate phrase-based models into our implementation, without compromising the  
658 recognition time. We also extend the recognizer with named entity models. These models attempt  
659 to improve recognition for proper nouns by estimating their pronunciation and language behavior.  
660 We exploit the fact that many named entities remain unchanged during English-to-Dutch translation  
661 implying that we can make reliable estimates for relevant named entities based on the source language  
662 text. Experiments show that the combination of phrase-based translation models and named entity  
663 models further reduces the recognition error to ca. 6.5% absolute and 25% relative on the same spoken  
664 Dutch translations from English. Moreover, the extensions come with the same efficiency benefits as  
665 the word-based model which allows their use in a real-time CAT environment. To our knowledge this  
666 is the first MT-based language model adaptation technique using a phrase-based translation model.  
667 More information can be found in [68].

## 668 6.2. Automatic Domain Adaptation

669 We also investigate the effect of automatic domain adaptation for speech recognition. We study  
670 both cross-domain adaptation and within-domain adaptation: the first approach adapts a model  
671 trained on a specific domain to other domains, while the second approach adapts to the current topic  
672 of the text.

673 For cross-domain adaptation, we chose to create a new data set. Previous recognition experiments  
674 were always performed on spoken translations of literature for which the domain is not always very  
675 confined. For this task we instead chose to work with 14 documentaries provided by VRT,<sup>14</sup> all of  
676 which have a specific domain i.e. mostly fauna and flora. For these data we have the following parallel  
677 data streams: (1) audio in English (original), (2) script in English (original), (3) audio in Dutch (voice  
678 over), (4) script in Dutch (as input for audio in Dutch), and (5) subtitles for the deaf in Dutch.

679 The audio is converted to the correct format and background noise is filtered out as much as  
680 possible. Subtitles are normalized to generate a ground truth transcription which is aligned with the  
681 audio to produce the necessary timing information. Baseline experiments with models that do not  
682 employ any domain adaptation yield acceptable word error rates, ranging from 9% to 33%.

683 In a first attempt we investigate two methods of exploiting domain knowledge: (1) fully  
684 automatic terminology extraction; (2) user-guided terminology extraction. The first method uses  
685 BiLDA to automatically extract relevant Dutch terminology based on the English translation. In the  
686 second approach, we develop semi-automatic methods in which the user/translator enters a Dutch  
687 query/description of the translation task. This query is then used to retrieve relevant terminology,  
688 using one of the following methods:

- 689 1. Word-to-word similarity based on a Latent Semantic Analysis (LSA) model [69]
- 690 2. Word-to-word similarity based on a continuous skip-gram model [70]
- 691 3. Document-to-document similarity based on LSA, followed by extraction of the most relevant  
692 words from the best matching document.

693 These methods are incorporated into the SCALE toolkit, which is described in [71]. Each of the  
694 investigated methods is first evaluated on text: for each documentary, the extracted terminology is  
695 compared to out-of-vocabulary (OOV) words: the most promising method is the one that is able to  
696 retrieve the most OOV words. In a next step, this terminology is added to the pronunciation lexicon  
697 and language model of the speech recognizer, and the word error rate of the domain-adapted speech

---

<sup>14</sup> VRT is the Flemish public broadcaster, cf. <https://www.vrt.be/en/>.

698 recognizer is measured. None of the proposed methods is able to extract enough relevant terminology  
699 consistently. Hence, we focus on other adaptation techniques. Moreover, we move from  $n$ -gram  
700 language models to the state-of-the-art RNNS LMs, more specifically long short-term memory (LSTM)  
701 [72].

702 A new topic of investigation for cross-domain adaptation is improving the modelling of OOV  
703 words. These are words that are not part of the speech recognizer's vocabulary and therefore cannot  
704 be recognized. OOV words are a known issue in cross-domain settings as the change of domain often  
705 introduces many domain-specific words. We work on combining word and character information in  
706 the LM, rather than only using word information. By using character information, the LM should be  
707 better able to see similarities between formally/morphologically similar words. This improves the  
708 quality of the model and reduces the number of parameters to train, because the vocabulary size when  
709 using characters is very small compared to words. Moreover, the model is better able to predict words  
710 following out-of-vocabulary words, because it can make use of the characters in the OOV word. Not  
711 only does our model improve on the existing language model, it also reduces its size. These findings  
712 are reported in [73]. The code for both baseline LSTM LMs and the word-character LSTM LMs is  
713 described in [74].

714 With respect to *within-domain adaptation*, we investigate three approaches. The first approach  
715 exploits the history, by combining the baseline LM with a continuous bag-of-words (CBOW) [70]  
716 representation of the previous words. We investigate how word embeddings are optimally combined  
717 into a history representation (e.g. mean, weighted mean or filtered mean) and how the resulting  
718 CBOW should be combined with the baseline RNN (at the input layer or the output layer of the RNN).  
719 Unfortunately, the improvements for small LMs did not extrapolate to larger LMs.

720 The second approach is similar to the CBOW model in that it builds a continuous representation  
721 of the history. However, rather than using word embeddings, the model uses an RNN to learn the  
722 weights of a fixed set of topics which were pretrained using Latent Dirichlet Allocation [75]. Using  
723 the weighted sum of these topics, the model should be able to predict topical words which can be  
724 combined with the baseline RNN LM. The results for this model are similar to the previous one: only  
725 improvements for smaller LMs are found. These findings are reported in [76].

726 The third model is a neural cache LM [77]. A cache model [78] is inspired by the fact that  
727 people tend to talk about the same topic for a while, such that words that have been used before in a  
728 conversation have a higher probability of being used again. In a neural cache LM, the previous words  
729 and their hidden representations are stored in a cache. A probability for the next word is calculated  
730 based on the similarity between the hidden representation of the current word and the representations  
731 stored in the cache. That cache probability is combined with the standard LM probability. We extend  
732 the neural cache model by starting from the intuition that a cache makes more sense for content words  
733 (e.g. *bilingual*, *backhand*) than for function words (e.g. *the*, *on*). We observe perplexity improvements  
734 by using the *information weight* of a word, which is large for content words and small for function  
735 words. We use the information weights to combine the cache and LM probability and to select which  
736 words should be added to the cache. Additionally, we compare the regular cache [78] and the neural  
737 cache [77] for speech recognition, and we find that, contrary to the results for perplexity, the regular  
738 cache performs better. The results of this research can be found in [79].

### 739 6.3. Translation of spoken data

740 In this section we focus on punctuation and segmentation insertion, since this is an important  
741 task for speech translation. Most ASR systems generate an output stream of words, which does not  
742 contain punctuation nor segmentation, apart from some form of acoustic segmentation which splits a  
743 transcript into so called utterances [80]. As these utterances may be very long and can contain several  
744 sentences, they are very hard to translate using MT engines. We tackle this issue in two steps: firstly,  
745 punctuation prediction and secondly, segmentation prediction.

Most MT engines are trained on data that contain punctuation marks. As the output of a speech recognition system usually contains no punctuation information, a solution needs to be found for this mismatch. We investigate several approaches.

**LM/LSTM based approaches** – One of the commonly used methods for inserting punctuation marks into ASR output is using a language model. Using an  $n$ -gram LM for punctuation insertion, without acoustic cues, can be considered to be the baseline of baselines. We also investigate the use of state-of-art LSTM LMs and additionally, LSTMs that are trained for sequence labeling. This means that we do not predict the next token at every time step as LMs do, but we predict whether the current word should be followed by a punctuation symbol or not. The last method is specifically trained for punctuation prediction and greatly reduces the number of possible output classes - from the whole vocabulary to the set of punctuation symbols and a symbol indicating ‘no punctuation’.

**Monolingual translation** – Peitz et al. [81] show improvements in BLEU score when using a monolingual translation system to translate from unpunctuated to punctuated text instead of an LM-based punctuation prediction method. They also do a system combination of hypotheses from different approaches, and get an additional improvement in BLEU score. They assume correct sentence segmentation.

We train different configurations of monolingual MT systems from non-punctuated Dutch to punctuated Dutch (to be used before the regular Dutch to English MT system), from non-punctuated Dutch to punctuated English, from non-punctuated Dutch to non-punctuated English, from punctuated Dutch to punctuated English, and from non-punctuated English to punctuated English. When we take the best configurations of each of these systems, we can measure total MT quality from unpunctuated Dutch to punctuated English in different conditions, as shown in Figure 11.

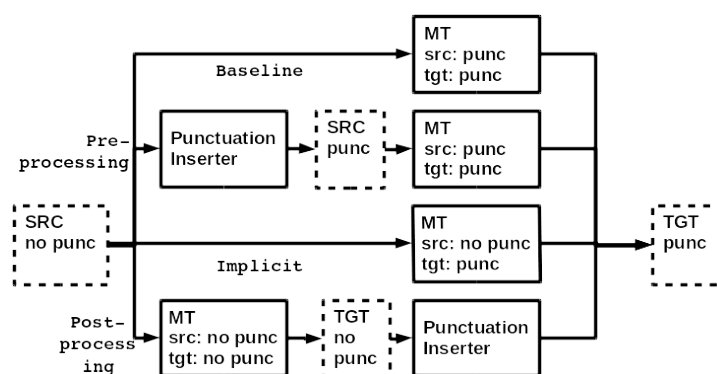


Figure 11. The different punctuation prediction strategies in a translation context.

In the **Baseline** we translate unpunctuated Dutch with the regular (punctuated) Dutch-English MT engine. In **Preprocessing**, we translate unpunctuated Dutch to punctuated Dutch, and take that output and translate it to English using the regular Dutch-English MT engine. In **Implicit Punctuation**, we translate unpunctuated Dutch to punctuated English using an MT system trained on unpunctuated Dutch as source and normal, punctuated English as target. In **Postprocessing**, we translate unpunctuated Dutch to unpunctuated English using an MT system trained on unpunctuated data for both languages. We take the output (unpunctuated English) and translate it to punctuated English using an MT engine trained on unpunctuated English to normal English.

Besides these different configurations, we also use different MT models: phrase-based and hierarchical SMT and neural MT. These MT paradigms are tested both for the monolingual systems and the bilingual systems. In total, by combining the  $n$ -gram LMs, LSTM LMs, LSTM sequence labeling, phrase-based SMT, hierarchical SMT and NMT as punctuation prediction models with the different configurations to insert the punctuation (pre-MT, during-MT or post-MT) and the three MT models for the actual translation, we tested 145 different experimental conditions. Since all setups are

782 trained and tested on the same data, this provides us a thorough comparison of punctuation prediction  
783 methods.

784 While there is a clear deterioration of MT quality when working with unpunctuated input, this  
785 gap can be closed for 66% in the case of our best MT system (NMT) by applying monolingual MT as  
786 punctuation insertion, or by using a dedicated implicit insertion MT system. Whether we use pre-  
787 or post-processing did not result in a significant difference, in most cases indicating that the general  
788 punctuation prediction quality for Dutch is similar to that of English. Full details are available in [82].

789 We also made some initial steps towards segmentation insertion. As MT systems work per  
790 segment (usually a sentence), the audio transcript is best divided into segments. This can be done  
791 based on auditory (length of pauses) or linguistic cues (lexical). Experimentation with different variants  
792 of these approaches will determine which is the most promising / best functioning approach.

## 793 7. The SCATE Interface

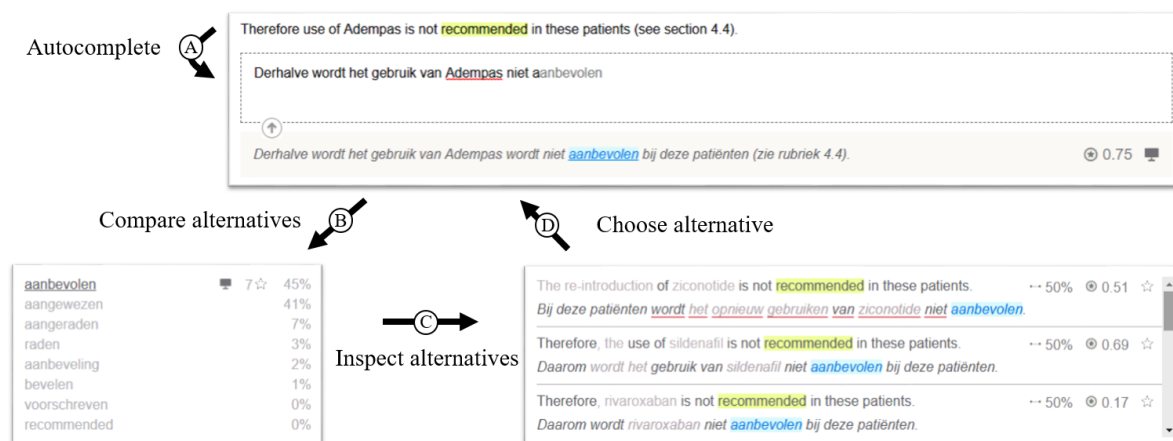
794 We use the studies of the translator's methods (section 5.1) to also investigate improvements  
795 to the translation environment's interface. The survey shows that ease of use is the most important  
796 motivation for choosing a translation environment, closely followed by speed of performance and  
797 features such as management of TMs and term bases. Contextual inquiries and interviews refine  
798 and enrich the insights on how translators work [3]. These studies form the start of a user-centered  
799 iterative research and development approach [83] resulting in a prototype translation environment.<sup>15</sup>  
800 To help translators understand how to use the rich set of translation suggestions that are offered in  
801 a translation environment, we paid specific attention to the intelligibility of these suggestions. The  
802 impact of intelligibility is explored in a comparative study with twenty-six professional translators.  
803 In a second round of evaluation, we invite four professional translators to compare our translation  
804 environment to Lilt [84].

### 805 7.1. Intelligible Translation Suggestions

806 The interface visualizes four established translation features: a term base that contains terms and  
807 their possible translation (in this case an automatically extracted term base, details of the extraction  
808 process are discussed by Coppers et al. [83]), fuzzy matches from a TM (Section 3.1), output of an  
809 MT system (Section 3.2), and auto-completion to predict a word or even a word group. In existing  
810 translation environments, such features often act like black boxes and provide only limited justification  
811 for their suggestions [85]. In order to improve trust [86], our interface explains where translation  
812 suggestions come from, in what context(s) they have been used before and how often they have been  
813 used by other translators (Figure 12). As a result, translators can make quick and well-informed  
814 decisions on the suitability of multiple alternatives in a particular translation context.

---

<sup>15</sup> A version with only cached results from the translation features is available at <http://scate.edm.uhasselt.be/>



**Figure 12.** All translation suggestions are closely related to each other. When a translator types a character, A) the auto-completion algorithm generates a suggestion. B) The translator compares this prediction to other alternatives. C) Interesting alternatives can be inspected in the context in which they have been used by other translators. D) When a translator decides which alternative to use, it can be added to the translation by pressing ENTER.

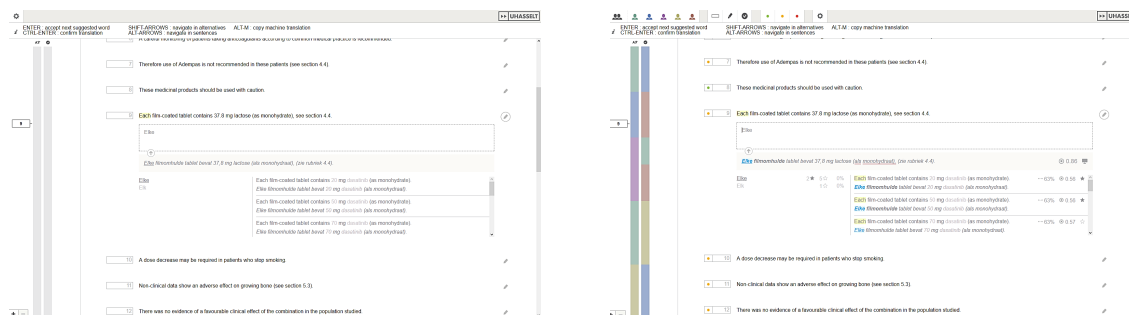
815 In order to efficiently combine sub-segments from various sources such as MT, TM and TB, the  
 816 SCATE interface contains an auto-completion feature that uses these sources to suggest (the remainder  
 817 of) a word or word group (Figure 12.A). By pressing ENTER, the translator can add this suggestion to  
 818 the translation. The algorithm justifies its prediction by selecting the suggestion in the sorted list of  
 819 alternatives (Figure 12.B), which shows several icons and metrics to explain where each alternative  
 820 comes from (e.g. MT, TM and TB) and how often it has been used before by other translators. The  
 821 occurrences themselves are highlighted in blue in the automatic translation and in the fuzzy matches  
 822 (Figure 12.C) to allow quick inspection of similar use cases. The automatic translation is shown very  
 823 close to the sentence to translate and stays directly available to the translator at any time (Figure 12).  
 824 As described in Section 3.2, parts in the automatic translations can be pretranslated by parts from the  
 825 fuzzy matches. This behavior is made clear to the translator by printing these parts in bold in the  
 826 automatic translation and in the matches they originate from.

827 Similar to other translation environments, the SCATE interface presents a similarity metric along  
 828 the fuzzy match to make clear how similar the sentence is. In contrast with existing environments, parts  
 829 that are similar according to the matching algorithm used are highlighted, rather than the differences.  
 830 Furthermore, the quality of each suggestion is determined by estimating the number of post-edits that  
 831 are still needed, using the technique described in Section 4.2.3. This estimate is normalized to a value  
 832 between 0 and 1, with 1 representing a score for a sentence that would not need post-editing. Parts of  
 833 the suggestion that probably require post-editing, are underlined in red. These visualisations help the  
 834 translator to quickly understand why a match was similar and how its translation might be useful.

835 As a result of the tight integration of suggestions from various sources, a translator can explore  
 836 up to four different relationships between suggestions at once: (1) the relationship between words  
 837 and word groups in the input sentence, (2) synonym recommendations, (3) source and destination  
 838 sentence in match recommendations and (4) the recommended automatic translation. As an additional  
 839 advantage, all translation aids require only limited space and can be combined into a compact  
 840 recommendation overview.

841 Keeping in mind the diversity in requirements during the study of translators' methods [3], the  
 842 interface allows translators to disable the additional metrics and highlighting according to their own  
 843 preferences. Figure 13b shows the interface with all explanations enabled whereas Figure 13a has  
 844 explanations disabled without compromising the functionality.

845 The prototype features end-user control over the workflow (Figure 14). A user can control whether  
 846 a segment requires a review, can be rejected by a reviewer, or can still be edited after confirmation.

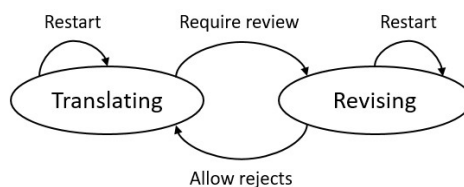


(a) The interface with all explanations disabled

(b) The interface with all explanations enabled

**Figure 13.** Two configurations of the SCATE interface with the same functionality.

847 Furthermore, it can be configured whether these transitions should be automated between phases,  
 848 and each segment can be assigned to a translator and reviser. By default, no such constraints are  
 849 enforced.

**Figure 14.** Each transition in the workflow is optional and can be enforced by the translation environment.

## 850 7.2. Evaluation

851 Section 7.2.1 describes an experiment measuring the effect of visualisations and intelligibility  
 852 features. Section 7.2.2 describes a user evaluation in which we compare the Lilt<sup>16</sup> and SCATE interfaces.

### 853 7.2.1. Influence of visualisation on experience and preference

854 To investigate the impact of intelligible translation aids on the translation process, we perform a  
 855 within subject user study with 26 professional translators. All participants translate two pieces of a  
 856 text of comparable difficulty using the two configurations of the SCATE interface shown in Figure 13.  
 857 The order in which they use each version of the interface is counterbalanced. After each condition  
 858 participants fill out a survey about the interface, with an additional comparative survey at the end.

859 The subjects are positive to very positive about both versions of the interface in the survey  
 860 questions. Analysis of the results shows that the visualisations help professional translators to assess  
 861 the quality of the generated suggestions and help to understand how these suggestions can be used  
 862 in translation, without distracting or negatively impacting efficiency. Intelligible visualisations do  
 863 not affect the quality of translation suggestions themselves, but instead inform translators about their  
 864 quality and context to support better decision making. Translators only prefer intelligible translation  
 865 aids when the additional information benefits the translation process, and when this information is  
 866 not yet part of the translator's readily available knowledge. Coppers et al. [83] provide more details  
 867 about the study.

### 868 7.2.2. Comparison with Lilt

869 In the second round of evaluation, we carry out a user study with four professional translators  
 870 and compare the SCATE prototype to Lilt, a commercial translation environment that stems from

<sup>16</sup> <http://www.lilt.com/>



871 research aiming to optimally combine human translation with MT [87]. The two systems under study  
872 can be considered as examples of a new generation of translation environment tools in the sense  
873 that they differ from the mainstream and most frequently used systems among translators such as  
874 SDL Trados Studio, Wordfast and memoQ in the following respects: (1) Both systems offer a tighter  
875 integration of MT and TM suggestions than the mainstream systems, giving MT a more prominent  
876 place. However, both systems adopt a fundamentally different approach to reach this goal. (2) Both  
877 systems present the active segment more centrally on the screen and the source and target text are  
878 presented vertically instead of horizontally in the standard view. Apart from that, they offer advanced  
879 user interaction features such as autocompletion and a variety of shortcuts to copy the different types  
880 of suggestions (TM, MT, alternative translations for words or fragments).

881 The interfaces of both translation environments share several aspects, such as the central  
882 placement of the active segment and the order in which source, target, automatic translation and  
883 alternatives are presented. When the experiment was carried out, the underlying MT architecture in  
884 both systems was SMT.<sup>17</sup> The main differences can be summarised as follows: (1) SCATE always shows  
885 suggestions from multiple sources, whereas Lilt offers these suggestions on demand. (2) Additional  
886 information about alternatives is displayed on the right-hand side of the user interface (memory search)  
887 in Lilt and is initially hidden, whereas this information is always present in the SCATE interface. (3)  
888 Lilt shows only one suggestion for the whole segment, while in SCATE the list of fuzzy matches is not  
889 limited to one. (4) Lilt uses adaptive MT while SCATE uses non-adaptive hybrid MT. (5) In SCATE,  
890 parts of suggestions, such as MT and fuzzy matches, can be used by double clicking individual words,  
891 which will add them to the translation.<sup>18</sup> (6) In SCATE, information is given about the source of the  
892 translation suggestions (hybrid MT, TM or term list) and additional scores are given (frequency, fuzzy  
893 match scores and a quality estimation score) whereas in the Lilt interface the source of the suggestion  
894 (TM or MT) can only be derived from the presence or absence of the fuzzy match percentage.

895 Four professional translators were paid for their participation (50 Euros per hour, 150 Euros in  
896 total) and signed an informed consent form. Prior to the experiment the participants were asked about  
897 their previous experience with translation and the use of translation environments. Next, they worked  
898 through a tutorial to become familiar with the user interface of either Lilt or SCATE, after which they  
899 translated a text 'for real' using the same interface. This first part was completed by a survey that  
900 asked about their experiences with the first environment. After that, they similarly worked through  
901 a tutorial, a translation session and a survey of the other interface. The experiment ended with a  
902 post-experiment survey reporting on their experience with the two interfaces.

903 As the SCATE prototype's MT component has exclusively been trained on English and Dutch  
904 medical texts, text selection for the experiment was also limited to medical material for this language  
905 combination. As none of the participants were experienced medical translators, text fragments  
906 were chosen from package leaflets intended for patients, on the assumption that these would be  
907 more manageable for the test subjects than highly technical texts. SCATE's corpus material is the  
908 English-and-Dutch EMEA TM as available through OPUS [88]. Although this is based on so-called  
909 EPARs (European Public Assessment Reports) rather than patient leaflets, both text types originate  
910 from the European Medicines Agency and share many features.

911 Care was taken to select texts on relatively new medicines that did not already feature in the  
912 EMEA TM. Two text fragments of equal size (175 words each) were prepared for the tutorials and  
913 two further fragments (225 and 232 words, 20 segments) were used for the actual translation. The  
914 test subjects' activities during translation were monitored using Inputlog [89] for keylogging and  
915 Camtasia<sup>19</sup> for screen recording. The order of texts and environments tested was balanced across  
916 participants. Although we could not fully control the Lilt environment, care was taken to create

---

<sup>17</sup> At the moment of writing, Lilt has replaced the SMT by NMT engines.

<sup>18</sup> Clicking once on any word will search for new alternative translations for that word.

<sup>19</sup> <https://www.techsmith.com/video-editor.html>

917 translation conditions that were as similar as possible. The same TM was used in both systems (198K  
 918 segments, 5.3 million words in total), and a manually created term list of 360 medical term pairs was  
 919 uploaded in both systems. To keep conditions stable across participants, we also hid SCATE's button  
 920 that enabled users to customise the interface (to turn features on/off).

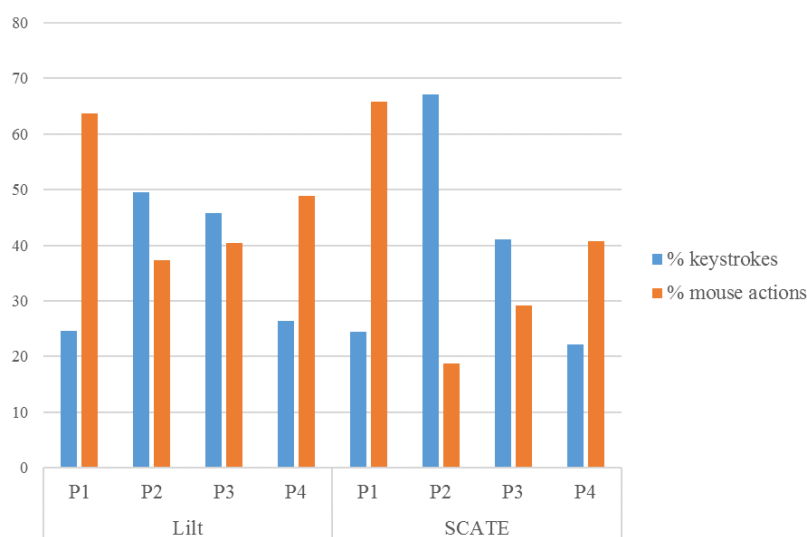
921 Table 2 gives an overview of the experiments that were carried out. The order of the two  
 922 environments and the two texts were balanced across participants. As the texts were of similar nature  
 923 and length, it was potentially interesting to check whether working in one interface rather than another  
 924 was faster or slower. A comparison of the total number of minutes spent per text per participant,  
 925 however, suggests that the difference seems to be more related to individual speed rather than to the  
 926 interface used, with P2 and P3 being faster in Lilt than P1 and P4, and P1, P3 and P4 having a similar  
 927 speed in SCATE.

Participant	Environment	Text	Total time	
P1	Exp1	Lilt	Text1	23
	Exp2	SCATE	Text2	19
P2	Exp1	Lilt	Text2	17
	Exp2	SCATE	Text1	14
P3	Exp1	SCATE	Text1	19
	Exp2	Lilt	Text2	15
P4	Exp1	SCATE	Text2	19
	Exp2	Lilt	Text1	27

**Table 2.** Per participant, the order of the experiments, the environment used, the text that was translated and the total time expressed in minutes.

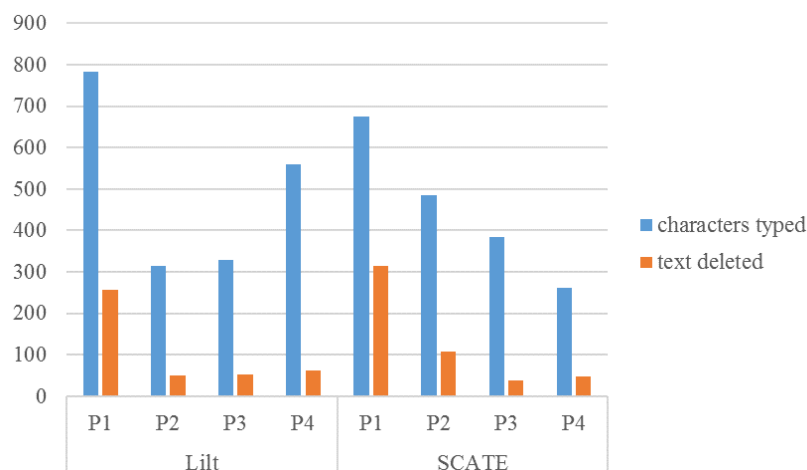
928 The study provides us with useful insights. Two translators (P2 and P3) started working on a  
 929 segment immediately after opening and combined different strategies: typing, inserting suggested  
 930 words as well as starting from the complete translation suggestion which they then revise or accept. The  
 931 two other translators (P1 in SCATE and P4 in both interfaces) preferred copying a complete translation  
 932 suggestion, (which could be either an MT or a TM suggestion) to the edit box to start from. One  
 933 translator (P4) pauses for a long time before she takes action. This finding is in line with [90], in which  
 934 two production styles were distinguished: translators either translate a segment mentally and then  
 935 type it (Prospective Thinking), or they translate as they were reading the text (Translating On-screen).  
 936 In all screen recordings we noticed that, despite the training phase, the individual strategies evolved  
 937 over time, a finding that was also reported by Koehn [91], in which a learning effect is described.

938 Figure 15 shows the percentage of time devoted to keystrokes versus mouse actions in both  
 939 translation environments. Again, individual differences can be observed. P1 has a noticeably higher  
 940 number of mouse actions than keystrokes, which is not surprising as she does not use any shortcuts in  
 941 either environment. P2 has a remarkably higher number of keystrokes in SCATE compared to Lilt,  
 942 which can be explained by his own comment in the survey that "in SCATE there was more typing  
 943 work when diverting from the original suggestion."



**Figure 15.** Percentage of time devoted to keystrokes vs. mouse actions per translation environment per participant.

944 Figure 16 presents the total number of characters typed versus the number of keys pressed to  
 945 delete text (delete, backspace). No distinction has been made between the typing activity in- or outside  
 946 the Lilt or SCATE environment. This figure demonstrates the benefits of using interactive translation  
 947 environments. Even P1, who produced most characters, only types around 700-800 characters to  
 948 translate a source document of 1300-1450 characters. A more drastic decrease can even be seen in P2  
 949 and P3 in Lilt and P3 and P4 in SCATE, with fewer than 400 characters typed.



**Figure 16.** Total number of characters typed versus text deleted per translation environment per participant.

950 To exemplify the minimal typing effort, figure 17 shows how P2 produced the translation '*Licht*  
 951 *uw arts in als u maag-of darmproblemen hebt (gehad)*'<sup>20</sup> in the SCATE environment. The letters in dark  
 952 blue are the characters that were actually typed; [RETURN] is used to insert/accept the suggested  
 953 word; [BACK] to delete characters and [CTRL+RETURN] to confirm the translation.

<sup>20</sup> Inform your doctor if you (have) had stomach or bowel problems.

```
{13339}[RSHIFT]L[RETURN][RETURN][RETURN][RETURN]
[RETURN][RETURN]{6680}pro[RETURN]hebt.[RSHIFT](gehad
[RSHIFT]).m[RETURN][RETURN][RETURN][RETURN][RETURN]
[BACK][BACK][BACK][BACK][BACK][BACK][BACK][BACK]
[BACK][BACK][BACK][BACK][BACK][BACK][BACK][BACK]
[BACK][BACK][BACK][BACK][BACK][BACK].[Movement][LEFT
Click]maag--of-darm{2168}[RCTRL][RCTRL + RETURN]
```

Figure 17. Example of how a translation is produced in SCATE.

954 Features of the new translation environment tools that were valued most by the participants  
 955 are the clean and calm design of the user interface of both systems, the interactive and adaptive MT  
 956 of Lilt and the frequency information of translation alternatives of SCATE. Translators find quality  
 957 estimation scores only useful when they are interpretable (the range of the scores should be clear) and  
 958 when they are in line with the more traditional fuzzy match scores that translators are acquainted with.  
 959 Translators would also like to know the origin of the suggestions (the difference between a TM or MT  
 960 suggestion was not clear in Lilt), and they find a concordance search indispensable. An 'undo'-button  
 961 would also be appreciated. The translators also raised concerns about the new interactive way of  
 962 translating as translators might be more inclined to produce translations word by word. Starting from  
 963 MT suggestions might have a negative impact on the overall readability of the text produced and  
 964 translators might become less focused when they are presented with good translations automatically.

965 Perhaps the most important conclusion of this study is that translators differ from each other in the  
 966 way they work. We observe individual preferences to interact with the system (shortcuts versus mouse)  
 967 and different ways of using the suggestions (copying the complete suggestion followed by revision or  
 968 gradually building up a translation by accepting appropriate suggestions) and it is important for CAT  
 969 tools to support these different styles of working.

970 Customisability of the user interface (a feature that we disabled to keep experimental conditions  
 971 stable) seems extremely important. This was also at the top wish list of the respondents in [4] to assess  
 972 the user interface needs of post-editors of MT.

## 973 8. Conclusion

974 We present an overview of the research that is performed in the SCATE project. We show the  
 975 coherence between several different aspects of our research, and how they all relate to the translator's  
 976 professional workflow. Although several aspects have been published before in isolation, this paper  
 977 provides the broader context, and presents additional research.

978 We describe how several aspects of the translation technologies can be improved, such as fuzzy  
 979 matching, integrating TM and MT technologies, and parallel treebanks for syntax-based MT. We are  
 980 convinced that acceptance of MT by the translator's community can grow through such an integration  
 981 of TM and MT.

982 We delve into quality estimation research on the word and sentence level, and as byproducts, we  
 983 built a taxonomy of MT errors and a corpus of manual post-editing and annotation of the MT errors  
 984 according to this taxonomy. These data allow to build informative quality estimation systems, not only  
 985 indicating what goes wrong, but also providing information on why this is the case.

986 We study translator's methods towards terminology extraction from comparable text and try out  
 987 different approaches to this problem, depending on the domain. We show that it is possible to do this  
 988 with only small supervision data sets.

989 We investigate several aspects of speech recognition in the context of translation, such as  
 990 post-editing through speech and automatic domain adaptation, where we show that speech recognition  
 991 can be improved by using information from within the translation engine, and by working on the  
 992 character level to solve the unknown word problem. We also performed an experiment to find out the  
 993 best approach towards punctuation insertion in speech translation.

994 Last but not least we present a user interface, based on user observation in practice, which  
995 provides a proper integration of many of the above described aspects into a convenient working  
996 environment using intelligible translation suggestions coming from several different sources. We set  
997 up an experiment evaluating this user interface, comparing it to an existing commercial interface.

998 All these research aspects show potential in improving the translator's daily workflow, not  
999 only implying an improved productivity, but also a customizable, more pleasant and calm, working  
1000 environment.

1001 **Author Contributions:** Conceptualization, Vincent Vandeghinste, Tom Vanallemeersch, Joris Pelemans, Lyan  
1002 Verwimp, Patrick Wambacq, Marie-Francine Moens, Els Lefever, Lieve Macken, Véronique Hoste, Jan Van den  
1003 Bergh and Kris Luyten; Funding acquisition, Vincent Vandeghinste, Tom Vanallemeersch and Frank Van Eynde;  
1004 Investigation, Vincent Vandeghinste, Tom Vanallemeersch, Liesbeth Augustinus, Bram Bulté, Joris Pelemans, Lyan  
1005 Verwimp, Geert Heyman, Iulianna van der Lek-Ciudin, Ayla Rigouts Terry, Arda Tezcan, Lieve Macken, Joke  
1006 Daems, Joost Buysschaert, Sven Coppers and Jan Van den Bergh; Methodology, Vincent Vandeghinste; Project  
1007 administration, Vincent Vandeghinste; Supervision, Vincent Vandeghinste, Frank Van Eynde, Patrick Wambacq,  
1008 Marie-Francine Moens, Frieda Steurs, Els Lefever, Lieve Macken, Véronique Hoste and Kris Luyten; Visualization,  
1009 Sven Coppers and Jan Van den Bergh; Writing – original draft, Vincent Vandeghinste, Tom Vanallemeersch, Bram  
1010 Bulté, Lyan Verwimp, Geert Heyman, Iulianna van der Lek-Ciudin, Arda Tezcan, Lieve Macken, Joke Daems,  
1011 Joost Buysschaert and Sven Coppers; Writing – review & editing, Vincent Vandeghinste, Patrick Wambacq and  
1012 Sven Coppers.

1013 **Funding:** The research in this project was funded by the Flemish Agency for Innovation and Technology IWT,  
1014 project number 13007.

1015 **Acknowledgments:** We would like to thank the companies and organizations taking part in the Industrial  
1016 Advisory Committee of the SCATE project for their ideas, feedback, and cooperation. These are Clarivate,  
1017 CommArt International, CrossLang, ITP Europe, Mastervoice, Nuance, OneLiner, Televic, VRT Onderzoek en  
1018 Innovatie, Xplanation, Yamagata-Europe, Yazzoom. We would also like to thank the additional translators that  
1019 participated in our inquiries.

1020 **Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the  
1021 study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to  
1022 publish the results.

### 1023 **Abbreviations**

1024 The following abbreviations are used in this manuscript:

1025

API	Application programming interface
ASR	Automated Speech Recognition
BiLDA	Bilingual Latent Dirichlet Allocation
BLEU	BiLingual Evaluation Understudy
BLI	Bilingual Lexicon Induction
BWESG	Bilingual Word Embedding Skip Grams
CAT	Computer-aided Translation
C-BiLDA	Comparable Bilingual Latent Dirichlet Allocation
CBOW	Continuous Bag-of-Words
CLARIN	Common Language Resources Research Infrastructure
DGT	Directorate General for Translation
DNT	Do-Not-Translate
EN	English
GrETEL	Greedy Extraction of Trees for Empirical Linguistics
GRU	Gated Recurrent Unit
HTER	Human-targeted Translation Edit Rate
IAA	Inter-Annotator Agreement
ITP	Interactive Translation Prediction
LENG	Lexical Equivalent Node Grouping
LM	Language Model
LSA	Latent Semantic Analysis
LSTM	Long Short-term Memory
1026 METEOR	Metric for Evaluation of Translation with Explicit Ordering
ML	Machine Learning
MT	Machine Translation
NL	Dutch
NMT	Neural Machine Translation
OOV	Out-of-Vocabulary
PET	Post-Editing Time
PoS	Part-of-Speech
QE	Quality Estimation
RBMT	Rule-based Machine Translation
RNN	Recurrent Neural Network
SCATE	Smart Computer-Aided Translation Environment
SMT	Statistical Machine Translation
TAP	Think Aloud Protocol
TB	Term-Base
TBX	Term-Base eXchange
TEnT	Translation Environment
TER	Translation Edit Rate
TM	Translation Memory
UI	User Interface
VRT	Vlaamse Radio en Televisie
WMT	Workshop on Machine Translation
1027	
1028	1. Lagoudaki, E. Translation Memories Survey 2006. User's perceptions around TM use. <i>Translation and the</i>
1029	<i>Computer</i> , 28, London, England: ASLIB, 1-29, 2006.
1030	2. O'Brien, S.; O'Hagan, M.; Flanagan, M. Keeping an eye on the UI design of Translation Memory : How
1031	do translators use the "Concordance" feature ? 28th Annual Conference of the European Association of
1032	Cognitive Ergonomics, 2010, pp. 187-190.
1033	3. Van den Bergh, J.; Geurts, E.; Degraen, D.; Haesen, M.; van der Lek-Ciudin, I.; Coninx, K. Recommendations
1034	for Translation Environments to Improve Translators' Workflows. <i>Translating and the Computer</i> 37.
1035	<i>AsLing</i> , 2015.

- 1036 4. Moorkens, J.; O'Brien, S. Assessing User Interface Needs of Post-Editors of Machine Translation. *Human*  
1037 *Issues in Translation Technology* **2017**, pp. 109–130.
- 1038 5. Koehn, P.; Haddow, B. Interactive Assistance to Human Translators using Statistical Machine Translation  
1039 methods. MT Summit XII, 2009, pp. 1–8.
- 1040 6. Sanchis-Trilles, G.; Alabau, V.; Buck, C.; Carl, M.; Casacuberta, F.; García-Martínez, M.; Germann, U.;  
1041 González-Rubio, J.; Hill, R.L.; Koehn, P.; Leiva, L.A.; Mesa-Lao, B.; Ortiz-Martínez, D.; Saint-Amand, H.;  
1042 Tsoukala, C.; Vidal, E. Interactive translation prediction versus conventional post-editing in practice: a  
1043 study with the CasMaCat workbench. *Machine Translation* **2014**, pp. 217–235.
- 1044 7. Torregrosa Rivero, D.; Pérez-Ortiz, J.A.; Forcada, M.L. Comparative Human and Automatic Evaluation  
1045 of Glass-Box and Black-Box Approaches to Interactive Translation Prediction. *The Prague Bulletin of*  
1046 *Mathematical Linguistics* **2017-06**, 108. doi:10.1515/pralin-2017-0012.
- 1047 8. Green, S.; Wang, S.I.; Chuang, J.; Heer, J.; Schuster, S.; Manning, C.D. Human Effort and Machine  
1048 Learnability in Computer Aided Translation. Proceedings of the 2014 Conference on Empirical Methods in  
1049 Natural Language Processing (EMNLP); Association for Computational Linguistics: Doha, Qatar, 2014; pp.  
1050 1225–1236. doi:10.3115/v1/D14-1130.
- 1051 9. Zaretskaya, A. The Use of Machine Translation among Professional Translators. Proceedings of the  
1052 EXPERT Scientific and Technological Workshop; , 2015; pp. 1–12.
- 1053 10. Teixeira, C. The Impact of Metadata on Translator Performance: How Translators Work With Translation  
1054 Memories and Machine Translation. PhD thesis, Universitat Rovira i Virgili and Katholieke Universiteit  
1055 Leuven, 2014.
- 1056 11. Vieira, L.N.; Specia, L. A Review of Translation Tools from a Post-Editing Perspective. 3rd Joint EM+ /CNGL  
1057 Workshop Bringing MT to the User: Research Meets Translators (JEC), 2011, p. 33–42.
- 1058 12. Snover, M.; Madnani, N.; Dorr, B.; Schwartz, R. TER-Plus: paraphrase, semantic, and alignment  
1059 enhancements to Translation Edit Rate. *Machine Translation* **2009**, 23, 117–127.
- 1060 13. Levenshtein, V.I. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics*  
1061 *Doklady* **1966**, 10, 707–710.
- 1062 14. Vanallemeersch, T.; Vandeghinste, V. Assessing linguistically aware fuzzy matching in Translation  
1063 Memories. Proceedings of the 18th Annual Conference of the European Association for Machine  
1064 Translation, 2015, pp. 153–160.
- 1065 15. Steinberger, R.; Eisele, A.; Klocek, S.; Pilos, S.; Schlüter, P. DGT-TM: A freely available Translation Memory  
1066 in 22 languages. *CoRR* **2013**, abs/1309.5226, [1309.5226].
- 1067 16. Koehn, P.; Hoang, H.; Birch, A.; Callison-Burch, C.; Federico, M.; Bertoldi, N.; Cowan, B.; Shen, W.;  
1068 Moran, C.; Zens, R.; Dyer, C.; Bojar, O.; Constantin, A.; Herbst, E. Moses: Open Source Toolkit for  
1069 Statistical Machine Translation. Proceedings of the 45th Annual Meeting of the ACL. Interactive Poster  
1070 and Demonstration Sessions; Association for Computational Linguistics: Prague, Czech Republic, 2007; pp.  
1071 177–180.
- 1072 17. Klein, G.; Kim, Y.; Deng, Y.; Senellart, J.; Rush, A.M. OpenNMT: Open-Source Toolkit for Neural Machine  
1073 Translation. Proceedings of the 55th Annual Meeting of the ACL; , 2017. doi:10.18653/v1/P17-4012.
- 1074 18. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. BLEU: A Method for Automatic Evaluation of  
1075 Machine Translation. Proceedings of the 40th Annual Meeting on Association for Computational  
1076 Linguistics; Association for Computational Linguistics: Stroudsburg, PA, USA, 2002; pp. 311–318.  
1077 doi:10.3115/1073083.1073135.
- 1078 19. Lavie, A.; Agarwal, A. Meteor: An Automatic Metric for MT Evaluation with High Levels of Correlation  
1079 with Human Judgments. Proceedings of the Second Workshop on Statistical Machine Translation;  
1080 Association for Computational Linguistics: Stroudsburg, PA, USA, 2007; StatMT '07, pp. 228–231.
- 1081 20. Bulté, B.; Vanallemeersch, T.; Vandeghinste, V. M3TRA: integrating TM and MT for professional translators.  
1082 Proceedings of the 21st Annual Conference of the European Association for Machine Translation; , 2018.
- 1083 21. Kotzé, G.; Vandeghinste, V.; Martens, S.; Tiedemann, J. Large Aligned Treebanks for Syntax-based Machine  
1084 Translation. *Language Resources and Evaluation* **2016**, 51, 249–282.
- 1085 22. Koehn, P. Neural Machine Translation. *CoRR* **2017**, abs/1709.07809, [1709.07809].
- 1086 23. Vandeghinste, V.; Martens, S.; Kotzé, G.; Tiedemann, J.; Van den Bogaert, J.; De Smet, K.; Van Eynde, F.;  
1087 van Noord, G. Parse and Corpus-based Machine Translation. In *Essential Speech and Language Technology for*  
1088 *Dutch: Results by the STEVIN programme*; Spyns, P.; Odiijk, J., Eds.; Springer, 2013; pp. 305–319.

- 1089 24. Williams, P.; Sennrich, R.; Post, M.; Koehn, P. *Syntax-based Statistical Machine Translation*; Synthesis Lectures  
1090 on Human Language Technologies, Morgan & Claypool Publishers, 2016.
- 1091 25. Li, J.; Xiong, D.; Tu, Z.; Zhu, M.; Zhang, M.; Zhou, G. Modeling Source Syntax for Neural Machine  
1092 Translation. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics,  
1093 2017, Vol. 1, pp. 688–697.
- 1094 26. Eriguchi, A.; Tsuruoka, Y.; Cho, K. Learning to Parse and Translate Improves Neural Machine Translation.  
1095 Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017, Vol. 2, pp.  
1096 72–78.
- 1097 27. Aharoni, R.; Goldberg, Y. Towards String-To-Tree Neural Machine Translation. Proceedings of the 55th  
1098 Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Association  
1099 for Computational Linguistics, 2017, pp. 132–140. doi:10.18653/v1/P17-2021.
- 1100 28. Nadejde, M.; Reddy, S.; Sennrich, R.; Dwojak, T.; Junczys-Dowmunt, M.; Koehn, P.; Birch, A. Predicting  
1101 Target Language CCG Supertags Improves Neural Machine Translation. Proceedings of the Second  
1102 Conference on Machine Translation. Association for Computational Linguistics, 2017, pp. 68–79.  
1103 doi:10.18653/v1/W17-4707.
- 1104 29. Chen, X.; Liu, C.; Song, D. Tree-to-tree Neural Networks for Program Translation, 2018. 32nd Conference  
1105 on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.
- 1106 30. Och, F.; Ney, H. A Systematic Comparison of Various Statistical Alignment Models. *Computational*  
1107 *Linguistics* **2003**, *29*, 19–51.
- 1108 31. Macken, L. Analysis of Translational Correspondence in view of Sub-sentential Alignment. Proceedings of  
1109 the METIS-II Workshop on New Approaches to Machine Translation, 2007, pp. 97–105.
- 1110 32. Kotzé, G. Complementary Approaches to Tree Alignment: Combining Statistical and Rule-Based Methods.  
1111 PhD thesis, University of Groningen, 2013.
- 1112 33. Tiedemann, J. Lingua-Align: An Experimental Toolbox for Automatic Tree-to-Tree Alignment. Proceedings  
1113 of the Seventh International Conference on Language Resources and Evaluation LREC, 2010.
- 1114 34. Zhechev, V.; van Genabith, J. Maximising TM Performance through Sub-Tree Alignment and SMT.  
1115 Proceedings of the Ninth conference of the Association for Machine Translation in the Americas, 2010.
- 1116 35. Vanallemeersch, T. Data-driven Machine Translation using Semantic Tree Alignment. PhD thesis, KU  
1117 Leuven, 2017.
- 1118 36. Augustinus, L.; Vandeghinste, V.; Vanallemeersch, T. Poly-GrETEL: Cross-Lingual Example-based  
1119 Querying of Syntactic Constructions. Proceedings of the 10th Language Resources and Evaluation  
1120 Conference (LREC), 2016.
- 1121 37. Augustinus, L.; Vandeghinste, V.; Schuurman, I.; Van Eynde, F. GrETEL. A Tool for Example-Based  
1122 Treebank Mining. In *CLARIN in the Low Countries*; Odijk, J.; van Hessen, A., Eds.; Ubiquity Press: London,  
1123 2017; chapter 22, pp. 269–280.
- 1124 38. Vandeghinste, V.; Augustinus, L. "Making Large Treebanks Searchable. The SoNaR case.". Workshop on  
1125 Challenges in the Management of Large Corpora (CMLC-2), LREC, Reykjavík, 2014.
- 1126 39. Vanroy, B.; Vandeghinste, V.; Augustinus, L. Querying Large Treebanks : Benchmarking GrETEL Indexing.  
1127 *Computational Linguistics in the Netherlands Journal* **2017**, *7*, 145–166.
- 1128 40. Bojar, O.; Chatterjee, R.; Federmann, C.; Graham, Y.; Haddow, B.; Huck, M.; Jimeno Yepes, A.; Koehn, P.;  
1129 Logacheva, V.; Monz, C.; Negri, M.; Neveol, A.; Neves, M.; Popel, M.; Post, M.; Rubino, R.; Scarton, C.;  
1130 Specia, L.; Turchi, M.; Verspoor, K.; Zampieri, M. Findings of the 2016 Conference on Machine Translation.  
1131 Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers. Association  
1132 for Computational Linguistics, 2016, pp. 131–198. doi:10.18653/v1/W16-2301.
- 1133 41. Bojar, O.; Chatterjee, R.; Federmann, C.; Graham, Y.; Haddow, B.; Huang, S.; Huck, M.; Koehn, P.; Liu,  
1134 Q.; Logacheva, V.; Monz, C.; Negri, M.; Post, M.; Rubino, R.; Specia, L.; Turchi, M. Findings of the  
1135 2017 Conference on Machine Translation (WMT17). Proceedings of the Second Conference on Machine  
1136 Translation, Volume 2: Shared Task Papers; Association for Computational Linguistics: Copenhagen,  
1137 Denmark, 2017; pp. 169–214.
- 1138 42. Specia, L.; Blain, F.; Logacheva, V.; Astudillo, R.; Martins, A. Findings of the WMT 2018 Shared Task on  
1139 Quality Estimation. Proceedings of the Third Conference on Machine Translation: Shared Task Papers,  
1140 2018, pp. 689–709.



- 1141 43. Tezcan, A.; Hoste, V.; Macken, L. SCATE Taxonomy and Corpus of Machine Translation Errors. In *Trends in*  
1142 *E-tools and Resources for Translators and Interpreters*; Pastor, G.C.; Durán-Muñoz, I., Eds.; Brill | Rodopi, 2017;  
1143 Vol. 45, *Approaches to Translation Studies*, pp. 219–244.
- 1144 44. Macken, L.; De Clercq, O.; Paulussen, H. Dutch Parallel Corpus: A Balanced Copyright-Cleared Parallel  
1145 Corpus. *Meta: Journal des traducteurs / Meta: Translators' Journal* **2011**, *56*, 274–390.
- 1146 45. Tezcan, A.; Hoste, V.; Macken, L. Estimating Post-Editing Time Using a Gold-Standard Set of Machine  
1147 Translation Errors. *Computer Speech & Language* **2018**.
- 1148 46. Van Noord, G. At Last Parsing Is Now Operational. TALN06. Verbum Ex Machina. Actes de la 13e  
1149 conference sur le traitement automatique des langues naturelles, 2006, pp. 20–42.
- 1150 47. Martins, A.F.; Astudillo, R.F.; Hokamp, C.; Kepler, F. Unbabel's Participation in the WMT16 Word-Level  
1151 Translation Quality Estimation Shared Task. Proceedings of the First Conference on Machine Translation;  
1152 Association for Computational Linguistics: Berlin, Germany, 2016; pp. 806–811.
- 1153 48. Tezcan, A.; Hoste, V.; Macken, L. Detecting Grammatical Errors in Machine Translation Output Using  
1154 Dependency Parsing and Treebank Querying. *Baltic Journal of Modern Computing* **2016**, *4*, 203–217.
- 1155 49. Tezcan, A. Informative Quality Estimation of Machine Translation Output. PhD thesis, Ghent University,  
1156 2018.
- 1157 50. Beyer, H.; Holtzblatt, K. *Contextual Design: Defining Customer-Centered Systems*; Morgan Kaufmann  
1158 Publishers Inc.: San Francisco, CA, USA, 1997.
- 1159 51. Lewis, C. *Using the "thinking Aloud" Method in Cognitive Interface Design*; Research report, IBM T.J. Watson  
1160 Research Center, 1982.
- 1161 52. Steurs, F.; van der Lek-Ciudin, I. Report on human terminology extraction. Deliverable D3.1. Technical  
1162 report, 2016.
- 1163 53. Bowker, L. Productivity vs. Quality? A Pilot Study on the Impact of Translation Memory Systems.  
1164 *Localisation Focus* **2005**, *4*, 13–20.
- 1165 54. LeBlanc, M. Translators on translation memory (TM). Results of an ethnographic study in three translation  
1166 services and agencies. *T&I -The International Journal of Translation and Interpreting Research* **2013**, *5*.
- 1167 55. Delpech, E.M. Leveraging Comparable Corpora for Computer-assisted Translation. In *Comparable Corpora*  
1168 *and Computer-Assisted Translation*; John Wiley & Sons, Inc., Hoboken, NJ, USA, 2014.
- 1169 56. Bernardini, S.; Castagnoli, S. Corpora for translator education and translation practice. In *Topics in Language*  
1170 *Resources for Translation and Localisation*; John Benjamins, 2008; pp. 39–55.
- 1171 57. Blancafort, H.; Ulrich, A.X.; Heid, U.; Tatiana, S.C.; Gornostay, T.; Claude, K.A.L.; Méchoulam, C.; Daille,  
1172 B.; Sharoff, S. User-centred Views on Terminology Extraction Tools: Usage Scenarios and Integration into  
1173 MT and CAT Tools. *Translation Careers and Technologies: Convergence Points for the Future (TRALOGY)*;  
1174 , 2011.
- 1175 58. De Smet, W.; Moens, M.F. Cross-Language Linking of News Stories on the Web using Interlingual Topic  
1176 Modeling. Proceedings of the CIKM 2009 Workshop on Social Web Search and Mining (SWSM@CIKM),  
1177 2009, pp. 57–64.
- 1178 59. Heyman, G.; Vulić, I.; Moens, M.F. C-BiLDA Extracting Cross-lingual Topics from Non-Parallel Texts by  
1179 Distinguishing Shared from Unshared Content. *Data Mining and Knowledge Discovery* **2016**, *30*, 1299–1323.  
1180 doi:10.1007/s10618-015-0442-x.
- 1181 60. Vulić, I.; Moens, M.F. Bilingual Word Embeddings from Non-Parallel Document-Aligned Data Applied to  
1182 Bilingual Lexicon Induction. Proceedings of the 53rd Annual Meeting of the Association for Computational  
1183 Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short  
1184 Papers). Association for Computational Linguistics, 2015, pp. 719–725. doi:10.3115/v1/P15-2118.
- 1185 61. Vulić, I.; De Smet, W.; Moens, M.F. Identifying Word Translations from Comparable Corpora Using Latent  
1186 Topic Models. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:  
1187 Human Language Technologies (ACL-HLT), 2011, pp. 479–484.
- 1188 62. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases  
1189 and their Compositionality. Proceedings NIPS, 2013, pp. 3111–3119.
- 1190 63. Heyman, G.; Vulić, I.; Moens, M.F. Bilingual Lexicon Induction by Learning to Combine Word-Level and  
1191 Character-Level Representations. Proceedings of EACL, 2017, pp. 1085–1095.
- 1192 64. Heyman, G.; Vulić, I.; Moens, M.F. A Deep Learning Approach to Bilingual Lexicon Induction in the  
1193 Biomedical Domain. *BMC Bioinformatics*, pp. 1–15.

- 1194 65. Rigouts Terryn, A.; Hoste, V.; Lefever, E. A Gold Standard for Multilingual Automatic Term Extraction  
1195 from Comparable Corpora: Term Structure and Translation Equivalents. Proceedings of the 11th Language  
1196 Resources and Evaluation Conference (LREC), 2018.
- 1197 66. Rodríguez, L.; Reddy, A.M.; Ros, R.C. Efficient Integration of Translation and Speech Models in Dictation  
1198 Based Machine Aided Human Translation. Proceedings ICASSP, 2012, pp. 4949–4952.
- 1199 67. Pelemans, J.; Vanallemeersch, T.; Demuynck, K.; Van hamme, H.; Wambacq, P. Efficient Language Model  
1200 Adaptation for Automatic Speech Recognition of Spoken Translations. INTERSPEECH 2015, 16th Annual  
1201 Conference of the International Speech Communication Association, Dresden, Germany, September 6-10,  
1202 2015, pp. 2262–2266.
- 1203 68. Pelemans, J.; Vanallemeersch, T.; Demuynck, K.; Verwimp, L.; Van hamme, H.; Wambacq, P. Language  
1204 Model Adaptation for ASR of Spoken Translations Using Phrase-based Translation Models and Named  
1205 Entity Models. 2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP  
1206 2016, Shanghai, China, March 20-25, 2016, pp. 5985–5989. doi:10.1109/ICASSP.2016.7472826.
- 1207 69. Deerwester, S.; Dumais, S.T.; Furnas, G.W.; Landauer, T.K.; Harshman, R. Indexing by Latent Semantic  
1208 Analysis. *Journal of the American Society for Information Science* **1990**, *41*, 391–407.
- 1209 70. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space.  
1210 *arXiv: 1301.3781* **2013**.
- 1211 71. Pelemans, J.; Verwimp, L.; Demuynck, K.; Van hamme, H.; Wambacq, P. SCALE: A Scalable Language  
1212 Engineering Toolkit. Proceedings of the Tenth International Conference on Language Resources and  
1213 Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.
- 1214 72. Hochreiter, S.; Schmidhuber, J. Long Short-term Memory. *Neural Computation* **1997**, *9*, 1735–1780.
- 1215 73. Verwimp, L.; Pelemans, J.; Van hamme, H.; Wambacq, P. Character-Word LSTM Language Models.  
1216 Proceedings of the 15th Conference of the European Chapter of the Association for Computational  
1217 Linguistics, EACL 2017, Volume 1: Long Papers, Valencia, Spain, April 3-7, 2017, pp. 417–427.
- 1218 74. Verwimp, L.; Van hamme, H.; Wambacq, P. TF-LM: TensorFlow-based Language Modeling Toolkit.  
1219 Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018,  
1220 Miyazaki, Japan, May 7-12, 2018.
- 1221 75. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet Allocation. *Journal of Machine Learning Research* **2003**,  
1222 *3*, 993–1022.
- 1223 76. Boes, W.; Van Rompaey, R.; Verwimp, L.; Van hamme, H.; Wambacq, P. Domain Adaptation for LSTM  
1224 Language Models. Computational Linguistics in the Netherlands: Abstracts, CLIN 27, Leuven, Belgium,  
1225 2017.
- 1226 77. Grave, E.; Joulin, A.; Usunier, N. Improving Neural Language Models with a Continuous Cache.  
1227 Proceedings of International Conference on Learning Representations (ICLR), 2017.
- 1228 78. Kuhn, R.; Mori, R.D. A Cache-Based Natural Language Model for Speech Recognition. *IEEE Transactions*  
1229 *on Pattern Analysis and Machine Intelligence* **1990**, *12*, 570–583.
- 1230 79. Verwimp, L.; Pelemans, J.; Van hamme, H.; Wambacq, P. Information-Weighted Neural Cache Language  
1231 Models for ASR. Proceedings of the IEEE Workshop on Spoken Language Technology (SLT), 2018.
- 1232 80. Matusov, E.; Mauser, A.; Ney, H. Automatic Sentence Segmentation and Punctuation Prediction for  
1233 Spoken Language Translation. 2006 International Workshop on Spoken Language Translation, IWSLT  
1234 2006, Keihanna Science City, Kyoto, Japan, November 27-28, 2006, pp. 158–165.
- 1235 81. Peitz, S.; Freitag, M.; Mauser, A.; Ney, H. Modeling Punctuation Prediction as Machine Translation.  
1236 Proceedings IWSLT, 2011, pp. 238–245.
- 1237 82. Vandeghinste, V.; Verwimp, L.; Pelemans, J.; Wambacq, P. A Comparison of Different Punctuation  
1238 Prediction Approaches in a Translation Context. Proceedings of the Annual Conference of the European  
1239 Association for Machine Translation EAMT, Alicante, Spain, 2018.
- 1240 83. Coppers, S.; Van den Bergh, J.; Luyten, K.; Coninx, K.; van der Lek-Ciudin, I.; Vanallemeersch, T.;  
1241 Vandeghinste, V. Intellingo: An Intelligible Translation Environment. Proceedings of the 2018 CHI  
1242 Conference on Human Factors in Computing Systems; ACM: New York, NY, USA, 2018; CHI '18, pp.  
1243 524:1–524:13. doi:10.1145/3173574.3174098.
- 1244 84. Green, S.; Heer, J.; Manning, C.D. The Efficacy of Human Post-Editing for Language Translation.  
1245 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2013, pp.  
1246 439–448.

- 1247 85. Sinha, R.; Swearingen, K. The Role of Transparency in Recommender Systems. CHI '02 Extended Abstracts  
1248 on Human Factors in Computing Systems; ACM: New York, NY, USA, 2002; CHI EA '02, pp. 830–831.  
1249 doi:10.1145/506443.506619.
- 1250 86. Bellotti, V.; Edwards, K. Intelligibility and Accountability: Human Considerations in Context-Aware  
1251 Systems. *Human-Computer Interaction* **2001**, *16*, 193–212. doi:10.1207/S15327051HCI16234\_05.
- 1252 87. Green, S.; Chuang, J.; Heer, J.; Manning, C.D. Predictive Translation Memory: A Mixed-Initiative System  
1253 for Human Language Translation. Proceedings of the 27th annual ACM symposium on User Interface  
1254 Software and Technology. ACM, 2014, pp. 177–187.
- 1255 88. Tiedemann, J. News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces.  
1256 In *Recent Advances in Natural Language Processing*; Nicolov, N.; Bontcheva, K.; Angelova, G.; Mitkov, R.,  
1257 Eds.; John Benjamins, Amsterdam/Philadelphia: Borovets, Bulgaria, 2009; Vol. V, pp. 237–248.
- 1258 89. Leijten, M.; Van Waes, L. Keystroke Logging in Writing Research: Using Inputlog to Analyze and Visualize  
1259 Writing Processes. *Written Communication* **2013**, *30*, 358–392.
- 1260 90. Asadi, P.; Séguinot, C. Shortcuts, Strategies and General Patterns in a Process Study of Nine Professionals.  
1261 *Meta* **2005**, *50*, 522–547.
- 1262 91. Koehn, P. A process study of computer-aided translation. *Machine Translation* **2009**, *23*, 241–264.